



PhD thesis

**Predictive healthcare modelling via kernel-based machine learning**  
**Nwegbu, N.**

---

Full bibliographic citation: Nwegbu, N. 2023. Predictive healthcare modelling via kernel-based machine learning. PhD thesis Middlesex University

Year: 2023

Publisher: Middlesex University Research Repository

Available online: <https://repository.mdx.ac.uk/item/10xq8v>

---

Middlesex University Research Repository makes the University's research available electronically.

Copyright and moral rights to this work are retained by the author and/or other copyright owners unless otherwise stated. The work is supplied on the understanding that any use for commercial gain is strictly forbidden. A copy may be downloaded for personal, non-commercial, research or study without prior permission and without charge.

Works, including theses and research projects, may not be reproduced in any format or medium, or extensive quotations taken from them, or their content changed in any way, without first obtaining permission in writing from the copyright holder(s). They may not be sold or exploited commercially in any format or medium without the prior written permission of the copyright holder(s).

Full bibliographic details must be given when referring to, or quoting from full items including the author's name, the title of the work, publication details where relevant

(place, publisher, date), pagination, and for theses or dissertations the awarding institution, the degree type awarded, and the date of the award.

If you believe that any material held in the repository infringes copyright law, please contact the Repository Team at Middlesex University via the following email address: [repository@mdx.ac.uk](mailto:repository@mdx.ac.uk)

The item will be removed from the repository while any claim is being investigated.

See also repository copyright: re-use policy: <https://libguides.mdx.ac.uk/repository>

# Predictive Healthcare Modelling via Kernel-based Machine Learning



**Nnanyelugo Osita Nwegbu**

**2245436**

School of Science and Technology

Director of Studies : Dr. David Windridge

Supervisors : Dr. Xiaohong Gao

: Dr. Santosh Tirunagari

*A dissertation submitted to Middlesex University London  
in partial fulfilment of the requirements for  
the degree of Doctor of Philosophy*

March 17, 2023



# ABSTRACT

Predictive modelling of clinical data is fraught with challenges arising from the manner in which events are recorded. Firstly, the aggregated electronic health records (EHR) contain complementary information from multiple sources and are characterised as heterogeneous due to the disparate innate properties of their constituents. Secondly, patients typically fall ill at irregular intervals and experience dissimilar intervention trajectories. This results in irregularly sampled and uneven-length heterogeneous data, which poses a problem for standard multivariate tools. The alternative of feature extraction into equal-length vectors via methods like bag-of-words (BoW) potentially discards useful information.

This research proposes an approach based on a kernel framework, in which data is maintained in its native form: discrete sequences of symbols. Bespoke kernel functions derived from variants of edit distance between pairs of sequences may then be utilized in conjunction with support vector machines (SVM) to classify the data. The framework via multi-kernel learning (MKL) provides a principled way of addressing the problem of modelling heterogeneous EHR entities; thus, we can algebraically combine multiple base kernels derived from real-valued and categorical entities into a single model. It also provides a means to combine weak discriminative standalone kernels in order to achieve superior results.

The proposed method was evaluated in the context of a prediction task involving determining susceptible patients likely to succumb to type 2 diabetes following an earlier episode of elevated blood pressure of 130/80 mmHg. Kernels combined via multi-kernel learning achieved an F1-score of **0.96**, outperforming classification with SVM **0.63**, Logistic Regression **0.63**, Long Short Term Memory **0.61** and Multi-Layer Perceptron **0.54** applied to a BoW representation of the data. An F1-score of **0.91** was achieved by combining symbolic kernels with kernels derived from 11 real-valued test measurements. The findings also showed a higher F1-score of **0.93** was achieved in a similar heterogeneous combination of kernels derived from symbolic EHR and from a single test measure, ‘Serum bilirubin level’ (Read code 44E..00). In addition, as a means of external validation of the proposed framework, an F1-score of **0.97** was achieved with MKL on an external dataset.

The proposed approach is consequently able to overcome the limitations associated with feature-based classification in the context of clinical data.

# Acknowledgement

I would like to express my deepest appreciation to my supervisory team for their support and encouragement. Firstly, I am profoundly grateful to Dr. David Windridge for accepting me as a student. This work would have been impossible without his insightful guidance and constructive feedback. I am thankful for his invaluable patience and for persisting with me despite the challenges. My sincere thanks to Dr. Xiaohong Gao for the kind support and advice given. Likewise, I am grateful to Dr. Santosh Tirunagari, whose timely arrival at the university has been of immense benefit to me. I would also like to thank Dr. Norman Poh for helping me settle in at the onset of this endeavour. I am equally grateful to my academic referees, Prof. C.M.I. Okoye, Prof. Mrs. F.N. Okeke, and Prof. C.E. Okeke.

I am extremely grateful to the Clinical Informatics and Health Outcomes Group at the University of Surrey, where I had a brief stint as a visiting research fellow. I would like to thank Prof. Simon de Lusignan and his team. Especially helpful to me during this time were Dr. Andy Mcgovern, Dr. M. D. Feher, Dr. Neil M. Munro, Jeremy van Vlymen, and Will Hinton. They introduced me to type 2 diabetes, epidemiology, and informatics.

I am deeply indebted to my late friend Pharaoh Okadigbo for his unwavering support and encouragement. We had imagined what the finish line would be like but for his untimely passing last year. I also appreciate the late Prof. Mrs. Ikejiani Clark's kindness and relentless encouragement to pursue this degree.

I would also like to extend my sincere gratitude to my friends Chukwudi Ononogbu, Ikenna Ononogbu, Ikenna Nze, Emeka Obi, Dayo Adeyemi, Obinna Nwokedi, Okechukwu Ngoka, Johnson Abugu, and Solomon Ezekiel. Thank you, guys, for your moral and financial support. Not to forget my sister-in-law, Mrs. Obioma Aja, who kept reminding me of my unmet goal until I finally registered. I am truly grateful for your support throughout this journey.

Lastly, I am thankful to my family: my boys (Kodi and Nonso), my siblings (Chinwe, Ikenna, Chidi, and Madu), and my parents (Mr. B. O. Nwegbu and Dr. Mrs. M. N. Nwegbu). Thank you for the support and encouragement. You provided the inspiration and motivation to persist even with the odds stacked heavily against me.

# Acronyms

**ANN** Artificial Neural Networks

**ARRA** American Recovery and Reinvestment Act

**ART** Adaptive Recursive Tree

**BoW** Bag-of-Words

**CER** Comparative Effectiveness Research

**CHF** Congestive Heart Failure

**COPD** Chronic Obstructive Pulmonary Disease

**CQRS** Quality Reporting Service

**CT** Computerised Tomography

**CTV3** Clinical Terms Version 3

**DFT** Discrete Fourier Transform

**DTW** Dynamic Time Warp

**DWT** Discrete Wavelets Transform

**ERP** Edit Distance with Real Penalty

**EDR** Edit Distance on Real Sequence

**EHR** Electronic Health Records

**EMR** Electronic Medical Records

**EPR** Electronic Patient Record

**FMD** Falsified Medicines Directive

**GLL** Generalized Local Learning

**GP** General Practice

**GPES** General Practice Extraction Service

**GPR** Gaussian Process Regression

**GPSoC** General Practice System of Choice

**HITECH** Health Information Technology for Economic and Clinical Health Act

**ICD** International Classification of Diseases

**LOOCV** Leave One Out Cross-Validation

**MAR** Missing at Random

**MCAR** Missing Completely at Random

**MED** Markov Edit Distance

**MRFED** Markov Random Field-based Edit Distance

**MKL** Multi-Kernel Learning

**MNAR** Missing Not at Random

**MRI** Magnetic Resonance Imaging

**NED** Normalized Edit Distance

**NLP** Natural Language Processing

**PAA** Piecewise Aggregate Approximation

**PET** Positron Emission Tomography

**PSD** Positive Semi-definite

**QoF** Quality Outcomes Framework

**RCT** Randomized Controlled Trial

**RKHS** Reproducing Kernel Hilbert Space

**RFE** Recursive Feature Elimination

**SAX** Symbolic Aggregate approXimation

**SHFM** Seattle Heart Failure Model

**SNOMED CT** Systematised Nomenclature for Medicine - Clinical Terms

**SVM** Support Vector Machines

**TFIDF** Term Frequency-Inverse Document Frequency



# Contents

Abstract . . . . .	i
Acknowledgement . . . . .	iii
<b>List of Figures</b>	<b>ix</b>
<b>List of Tables</b>	<b>xii</b>
<b>1 Introduction</b>	<b>2</b>
1.1 Background . . . . .	2
1.2 Gaps Identified . . . . .	6
1.3 Proposed Research . . . . .	7
1.3.1 Research Question . . . . .	7
1.3.2 Aim of the Research . . . . .	7
1.3.3 Research Objectives . . . . .	7
1.3.4 Specific problem to be addressed . . . . .	7
1.3.5 Rationale . . . . .	8
1.4 Contributions . . . . .	8
1.5 Expected Impact . . . . .	9
1.6 Proposed Kernel Framework . . . . .	9
1.7 Advantages of the Kernel Framework . . . . .	10
1.8 Published Material . . . . .	10
1.9 Type 2 Diabetes Mellitus . . . . .	10
1.10 Dissertation Layout . . . . .	13
<b>2 Proposed Kernel Framework</b>	<b>14</b>
2.1 Linear Functions . . . . .	14
2.2 The Learning Problem . . . . .	14
2.3 Hyperplane Classifiers . . . . .	14
2.4 Kernel Definition . . . . .	15
2.4.1 Positive definiteness . . . . .	16
2.4.2 Inner product space - (Hilbert Space) . . . . .	16
2.4.3 Mercer's Theorem . . . . .	17
2.4.4 Reproducing Kernel Hilbert Space (RKHS) . . . . .	18
2.4.5 Cauchy-Schwarz Inequality for kernels . . . . .	19
2.4.6 Metric Spaces . . . . .	19
2.4.7 Conditionally positive semi definiteness . . . . .	19
2.4.8 Indefinite kernels . . . . .	20
2.4.9 Krein Space . . . . .	20
2.4.10 Spectral decomposition of Indefinite kernels . . . . .	20
2.4.11 Additional kernel construction methods . . . . .	21
2.4.12 Normalizing a kernel . . . . .	22

2.5	Summary	22
<b>3</b>	<b>Literature Review</b>	<b>23</b>
3.1	Electronic Health Records (EHR)	23
3.2	Multivariate Analysis with EHR	27
3.3	Problems with Modelling EHR Data	30
3.3.1	Heterogeneous EHR data	30
3.3.2	Irregularly sampled EHR data	32
3.3.3	Others	35
3.4	Elastic Distance Measures	36
3.4.1	Edit distance	37
3.4.2	Edit kernels	42
3.4.3	Indefinite kernels	46
3.5	Multi-Kernel Learning	47
3.6	Summary	49
<b>4</b>	<b>Methodology</b>	<b>51</b>
4.1	Overall Study Design	51
4.1.1	Independent variables	51
4.1.2	Dependent variables	51
4.2	Multivariate Analysis	52
4.2.1	Edit distance and variants	52
4.2.2	Example derivation of edit distance	55
4.3	Proposed Kernels	56
4.3.1	Proposed kernel construction methods	57
4.3.2	Spectral modification	60
4.4	Kernel-based Learning Algorithms	61
4.4.1	Support Vector Machine	61
4.4.2	Gaussian Process	62
4.5	Multi-Kernel Learning (SimpleMKL)	63
4.6	Kernel Evaluation and Selection	64
4.6.1	Kernel Alignment	64
4.6.2	Spectral Ratio	64
4.6.3	Classification performance and evaluation	65
4.7	Baseline Models	65
4.7.1	Baseline kernel functions	66
4.8	Dataset	66
4.8.1	Data preprocessing/cleansing	67
4.8.2	Reference validation dataset	70
4.9	Summary	71
<b>5</b>	<b>Experiments/Results</b>	<b>72</b>
5.1	Description	72
5.2	Experimental Objectives	72
5.3	Results / Findings	74
5.3.1	Effect of different kernel functions	74
5.3.2	Edit distance on sequences with common items	77
5.3.3	Single vs multiple kernels:	79
5.3.4	Edit distance on variable-length sequences:	81
5.3.5	Multiple kernel learning of heterogeneous entities:	85

5.3.6	Measure predictive performance of the data tables . . . . .	88
5.3.7	Static evaluation of kernels . . . . .	90
5.3.8	Comparison with traditional bag-of-words (BOW) . . . . .	93
5.3.9	Validation on external data . . . . .	95
5.4	Summary . . . . .	97
<b>6</b>	<b>Discussion</b>	<b>99</b>
6.1	Related Work . . . . .	102
6.2	Future Work . . . . .	103
6.3	Limitations . . . . .	103
6.4	Conclusion . . . . .	103
6.5	Summary . . . . .	104
<b>A</b>	<b>Raw results</b>	<b>105</b>
<b>B</b>	<b>Sample Raw Data Extracts</b>	<b>116</b>
B.1	Clinical Table . . . . .	116
B.2	Recall Table . . . . .	118
B.3	Refer Table . . . . .	119
B.4	Repeat Table . . . . .	121
B.5	Test Table . . . . .	123
B.6	Therapy Table . . . . .	125
<b>C</b>	<b>Kernel Evaluation</b>	<b>127</b>
	<b>References</b>	<b>133</b>

# List of Figures

1.1	The electronic health record (EHR) of a patient can be viewed as a repository of information regarding his or her health status in a computer-readable form. An encounter with the health-care system generates various types of patient-linked data. In the example shown, a heterogeneous mixture of medication, laboratory, imaging and narrative data are all generated [126] . . . . .	3
1.2	Example showing irregular intervals between longitudinal uneven-length blood pressure data for three patients . . . . .	3
1.3	Flat-file tabular representation of EHR data. Each row represents a patient and each column contains a variable. This is further processed into features by discarding some variables. . . . .	4
1.4	An illustration depicting entities with diverse physical characteristics that make up heterogeneous EHR data. . . . .	8
4.1	The kernel learning pipeline . . . . .	51
4.2	The proposed kernel framework for disease prognosis modelling with EHR data . . . .	52
4.3	Edit distance applied to both sequence of symbols . . . . .	56
4.4	Dynamic programming approach used to calculate the Edit distance of both sequences . . . . .	56
4.5	Evaluating the pairwise kernel function by first extracting the data as a sequence of symbols, then computing the edit distance between a pair of sequences. An edit cost of 2 is applied where symbols are <b>substituted</b> while 1 is applied if a symbol is <b>deleted</b> or <b>inserted</b> . The total cost is computed and used to derive the kernel function value as specified in Equation 4.27 . . . . .	60
4.6	The multi-kernel learning (MKL) framework for combining kernels derived from disparate data types . . . . .	63
4.7	Plots showing the uneven sequence length distribution and the aggregate mean, maximum and minimum length distribution according to the datasets . . . . .	67
4.8	Plot showing distribution of length of sequences per data table . . . . .	68
4.9	Distribution of number of instances of blood pressure, Health education offered, weight and cardiac disease monitoring recorded against patient records . . . . .	70
5.1	Comparison of the best performance achieved per kernel construction method and applied to Clinical, Recall, Refer, Repeat, Test, Therapy and All data datasets. . . . .	78
5.2	F1-Score and Accuracy achieved with distance substitution kernel construction method applied to the datasets . . . . .	78
5.3	F1-score and accuracy achieved with 7 kernel construction methods via 5 edit distance variants applied to the single view dataset (Alldata) . . . . .	78

5.4	F1-Score and Accuracy achieved by applying the kernels on symbols common to the pair of sequences via native ‘edit distance’ ( $d_{ed1}$ ), ‘edit distance normalised by the length of the longer sequence’ ( $d_{ed2}$ ), ‘edit distance normalised by the the number of common items’ ( $d_{ed3}$ ), and ‘edit distance normalised by the exponent of number of common items’ ( $d_{ed4}$ ) . . . . .	79
5.5	Best F1-Score and Accuracy achieved with a single kernel vs MKL using the distance substitution kernel construction method. This compares results from stand-alone single kernels displayed in Table 5.7 against the MKL results in Table 5.10 . . . . .	81
5.6	Comparison of performance achieved with single kernel vs MKL. Where <b>Exp 1:</b> Compares MKL vs single kernels applied to common symbols from the single view dataset - see 5.15 where the MKL results are displayed. <b>Exp 2:</b> Refers to experiments with kernel derived from edit distance computed with number of matched and unmatched symbol ( $d_{ed9}$ ) applied to the single view dataset. See 5.16 for the MKL results. <b>Exp 3:</b> Experiments via Edit distance with controlled equality ( $d_{ed10}$ ) applied to the therapy dataset. See Table 5.17 . . . . .	81
5.7	MKL F1-Score achieved from sequence length variation experiment . . . . .	82
5.8	MKL coefficients learned from combining 36 Gaussian Edit kernels applied to common symbols and applied to the single view dataset. This corresponds to the MKL results displayed in Table 5.15 . . . . .	85
5.9	MKL coefficients learned from combining 45 Gaussian Edit distance computed with unmatched and matched symbol applied to alldata. This corresponds to the MKL results displayed in Table 5.16 . . . . .	85
5.10	MKL coefficients learned from combining 36 Edit kernels applied to Therapy dataset. This corresponds to the MKL results displayed in Table 5.17 . . . . .	86
5.11	Single kernel performance vs MKL with Heterogeneous combination of edit kernels applied to 11 numeric Test measurements and edit kernel with distance ( $d_{ed9}$ ) applied to single view data (All data) . . . . .	86
5.12	Distribution of MKL weight coefficients indicating SimpleMKL favours a sparse solution. These are the corresponding weights for the results in Figure 5.11 . . . . .	87
5.13	Best stand-alone single kernel performance vs MKL with heterogeneous combination of edit kernels applied 11 real-valued Test measurements and symbolic therapy dataset. . . . .	88
5.14	Distribution of MKL weight coefficients indicating SimpleMKL favours a sparse solution. These are the corresponding weights for the results in Figure 5.13 . . . . .	88
5.15	Single kernel performance vs MKL with Heterogeneous combination of edit kernels applied 11 numeric Test measurements and symbolic single view data using multiple edit distances. . . . .	89
5.16	Distribution of MKL weight coefficients indicating SimpleMKL favours a sparse solution . . . . .	89
5.17	Comparison of F1-score and Accuracy achieved with 3 MKL experiments conducted based on heterogeneous combination of real-valued and symbolic data. The results for each single test is displayed in addition to the results achieved with all 11 test measurements . . . . .	89
5.18	F1-Scores obtained with each example used as the zero vector in the MKL combination of kernels generated with the distance substitution kernel construction method applied to the six datasets. These correspond to the results displayed in Table 5.10 . . . . .	94
5.19	F1-Scores obtained with each example used as the zero vector in the MKL combination of kernels generated with the distance substitution kernel construction method applied to the ‘All data’ dataset. These correspond to the results displayed in Table 5.10 . . . . .	95

5.20	F1 score obtained by spectral transformation (clip, shift, flip and square) compared against indefinite kernels of the 36 Gaussian Edit kernels (common symbols) and applied to the single view dataset. . . . .	96
5.21	F1 score obtained by spectral transformation (clip, shift, flip and square) compared against indefinite kernels of the 45 Gaussian Edit kernels (distance computed with matched and unmatched symbols) applied to the single view dataset. . . . .	96
5.22	F1 score obtained by spectral transformation (clip, shift, flip and square) compared against 4 indefinite kernels out of the 36 Gaussian Edit kernels applied to the Therapy dataset. . . . .	97

# List of Tables

1.1	Sample view of categorical EHR data (see Appendix B for more examples)	5
1.2	Sample view of numeric EHR data (see Appendix B for more examples)	6
1.3	Sample view of medication data with numeric and categorical content (see Appendix B for more examples), where <b>Str:</b> Strength and <b>Qty:</b> Quantity	6
3.1	Capabilities accessible via the GP IT Futures Framework Capability	25
4.1	Table showing sample data extracted from the clinical table for patient 1	56
4.2	Table showing sample data extracted from the clinical table for patient 2	56
4.3	Edit kernel functions constructed with the formulations expressed in equations 4.14, 4.16, 4.18, 4.20, 4.22, and 4.24 in conjunction with variants of edit distances defined in section 4.2.1. Where <b>ED:</b> Edit Distance, <b>Pseu:</b> pseudo, <b>Sim:</b> Similarity, <b>Gaus:</b> Gaussian, <b>RQ:</b> Rational Quadratic, <b>Poly:</b> Polynomial, <b>DS:</b> Distance Substitution, <b>TM:</b> Template Matching	61
4.4	Details of the data tables showing their description, minimum, maximum, and mean length of the sequences	67
4.5	<b>Table showing sample data extracted from the clinical table for patient 1</b>	68
4.6	Table showing 11 Test Events from the Test dataset with numeric values. The Test Read code, Read term, description showing the normal range, unit of measure, and the mean (std) values extracted from the dataset are displayed	69
5.1	Best classification performance obtained with edit distance measures implemented as a pseudo kernels, where <b>Ker :</b> Kernel functions; <b>F1 :</b> F1-score; <b>Acc :</b> Accuracy; <b>Sen :</b> Sensitivity; <b>Spec :</b> Specificity; <b>Mod :</b> Spectrum Modification; <b>Trans :</b> post-kernel Transformation; <b>C :</b> SVM C parameter; <b>nSV :</b> Number of Support Vectors; <b>%-ve Eig :</b> Percentage -Number of negative Eigenvalues	74
5.2	Best classification results obtained with edit similarity kernels	75
5.3	Best classification performance achieved with the Gaussian kernel method	75
5.4	Best classification performance achieved with the rational quadratic kernel method	76
5.5	Best classification performance achieved with the polynomial kernel method	76
5.6	Best classification performance achieved with the template matching method. Where <b>X<sub>0</sub> :</b> Template candidate	76
5.7	Best classification performance achieved with the distance substitution method. Where <b>X<sub>0</sub> :</b> Zero Vector Example	77
5.8	Best classification performance displayed according to the kernel construction method	77
5.9	Best classification performance achieved with the kernel construction methods applied to common symbols	77
5.10	Best results obtained from MKL convex optimization combining the four kernels ( <b>k<sub>ds.ed1</sub></b> , <b>k<sub>ds.ed2</sub></b> , <b>k<sub>ds.ed3</sub></b> , and <b>k<sub>ds.ed4</sub></b> ) applied independently to the respective datasets	79

5.11 MKL results obtained from combining 24 kernel matrices derived from the datasets. (The 4 kernels - $\mathbf{k}_{ds\_ed1}$ , $\mathbf{k}_{ds\_ed2}$ , $\mathbf{k}_{ds\_ed3}$ , and $\mathbf{k}_{ds\_ed4}$ per dataset) . . . . .	80
5.12 Weights (sigma) obtained from multi kernel learning process with classification performance displayed in Table 5.10 . . . . .	80
5.13 Kernel Target Alignment scores obtained with the 4 kernels derived with the distance substitution methods applied to all datasets. This shows the alignment with the target labels for the results displayed in Table 5.10 . . . . .	80
5.14 Best MKL classification performance achieved by varying the length of the sequences.	83
5.15 MKL combination of Pseudo and Gaussian Edit kernels applied to common symbols from the single view dataset via the 4 distances ( $d_{ed1}$ , $d_{ed2}$ , $d_{ed3}$ , and $d_{ed4}$ ). The classification results displayed according to dataset sizes - 158, 134, 115, and 104 data points. Where <b>No</b> : Number of Kernels combined . . . . .	83
5.16 MKL with Gaussian ‘edit distance computed with unmatched and matched symbols’ ( $d_{ed9}$ ) applied to single view dataset (all data). The classification results are displayed according to dataset sizes - 158, 134, 115, and 104 data points . . . . .	83
5.17 MKL result obtained on the therapy data. Classification results displayed according to dataset sizes - 158, 134, 115, and 104 data points . . . . .	84
5.18 MKL with a combination of Gaussian Edit distance kernels computed on 30 numeric Test measurements. Classification results displayed according to dataset sizes - 158, 134, 115, and 104 data points . . . . .	84
5.19 MKL with a combination of 810 Gaussian edit distance kernels computed on 30 numeric Test measurements. Classification results displayed according to dataset sizes - 158, 134, 115, and 104 data points . . . . .	84
5.20 Corresponding learned MKL weights for results displayed on Table 5.18 . . . . .	85
5.21 Corresponding learned MKL weights for results displayed on Table 5.19 . . . . .	86
5.22 Distance substitution kernel assessment . . . . .	91
5.23 Static kernel assessment Summary . . . . .	92
5.24 Kernel goodness measure for kernels generated from Edit distance on sequences with common items in Figure 5.4 . . . . .	92
5.25 Kernel goodness measure for kernels generated using the template matching method with results displayed in Table 5.6 . . . . .	92
5.26 Kernel goodness measure for kernels generated multiple kernel learning of heterogeneous entities in section 5.3.5. The kernels implement the distance $d_{ed10}$ . . . . .	93
5.27 Kernel goodness measure for kernels with results displayed in Table 5.16 . . . . .	93
5.28 Kernel goodness measure for kernels with results displayed in Table 5.17 . . . . .	93
5.29 Performance result obtained with Logistic Regression and SVM applied to bag-of-words and binary bag-of-words feature representation of the data . . . . .	93
5.30 Performance result obtained with deep learning LSTM and MLP applied to bag-of-words and binary bag-of-words feature representation of the data . . . . .	94
5.31 Best results obtained from classification with single kernels applied to the validation (peptide) data . . . . .	97
5.32 Best MKL results obtained by combining the top 7 kernels applied to the validation (peptide) data . . . . .	97
5.33 Performance result obtained with deep learning LSTM and MLP applied to bag-of-words and Binary bag-of-words feature representation of the validation (peptide) data . . . . .	97
5.34 Performance result obtained with Logistic Regression and SVM applied to bag-of-words and Binary bag-of-words feature representation of the validation (peptide) data . . . . .	98



A.1	Table showing classification performance achieved with the pseudo kernels executed on various datasets . . . . .	106
A.2	Table showing classification performance achieved with the edit similarity kernels executed on multiple datasets . . . . .	107
A.3	Table showing classification performance achieved with the Gaussian edit kernels executed on multiple datasets . . . . .	108
A.4	Table showing classification performance achieved with the Rational Quadratic edit kernels executed on multiple datasets . . . . .	109
A.5	Table showing classification performance achieved with the polynomial edit kernels executed on multiple datasets . . . . .	110
A.6	Table showing classification performance achieved with the edit distance substitution kernels executed on various datasets . . . . .	110
A.7	Table showing classification performance achieved with the edit template matching kernels executed on various datasets . . . . .	111
A.8	MKL classification performance achieved by combining 12 edit kernels on Clinical Table	111
A.9	MKL classification performance achieved by combining 12 edit kernels on Recall Table	112
A.10	MKL classification performance achieved by combining 12 edit kernels on Refer Table	112
A.11	MKL classification performance achieved by combining 12 edit kernels on Repeat Table	113
A.12	MKL classification performance achieved by combining 12 edit kernels on Test Table .	113
A.13	MKL classification performance achieved by combining 12 edit kernels on Therapy Table	114
A.14	Table showing results obtained by executing the kernel functions on a pair of sequences with common symbols . . . . .	114
B.1	Clinical Table . . . . .	117
B.2	Recall Table . . . . .	119
B.3	Refer Table . . . . .	121
B.4	Repeat Table . . . . .	123
B.5	Test Table . . . . .	124
B.6	Therapy Table . . . . .	126
C.1	Static kernel assessment Clinical dataset . . . . .	127
C.2	Static kernel assessment Recall dataset . . . . .	128
C.3	Static kernel assessment Refer dataset . . . . .	129
C.4	Static kernel assessment Repeat dataset . . . . .	130
C.5	Static kernel assessment Test dataset . . . . .	130
C.6	Static kernel assessment Therapy dataset . . . . .	131
C.7	Static kernel assessment All data dataset . . . . .	132
C.8	Static kernel assessment (Common symbols) All data dataset . . . . .	132

# Chapter 1

## Introduction

### 1.1 Background

The ubiquitous use of General Practice (GP) computer systems offers healthcare practitioners a reliable tool in the management and delivery of care in the context of routine primary care. The departure from paper-based healthcare service delivery towards technology was driven by a combination of factors. First and foremost, it was necessitated by the need to reduce the spiralling overall cost of care, reduce medical errors, and improve health care [117]. These initial objectives were not easily met at the onset due to the exorbitant cost of deploying an IT infrastructure to support healthcare services without immediate financial returns [144]. As a result, the adoption rate was slow. Nevertheless, the need to modernise and digitize clinical workflows showed benefits such that government intervention played a pivotal role in driving up its use. In the UK, the defunct General Practice System of Choice (GPSoC) framework was used by the NHS to fund the deployment of technology to the local practices. It specified the minimum core operational requirements primary care IT systems providers had to meet, such as showing interoperability with third-party providers, among other key functionalities. In addition, it encouraged competition by granting local practices the freedom to choose from suppliers who met the conditions. A typical healthcare IT system is expected to provide the following basic functionalities as categorised into: electronic clinical information, computerised provider order entry, result management, and decision support [52]. The NHS is highly committed to technology infrastructure by appropriating an extra £2.4 billion a year to support GPs [180]. Consequently, technology has revolutionized healthcare management and enabled the accumulation of complementary information from various sources, such as secondary care units, hospitals, clinics, and laboratories, into a single repository. This provides primary care practitioners with a one-stop view of patient health trajectories, histories, and medications among other relevant entities that may be consulted before reaching a clinical decision.

A typical database often consists of disease symptoms, medical procedures, current and past medication, existing conditions and investigations, allergies, intolerances, test results from laboratories, ultrasonic and MR images, time series, etc (See Figure 1.1 for a depiction of a typical patient's electronic health record (EHR)). These are captured and stored in both structured and unstructured data formats. A coded dictionary of terminologies provides a means to record structured clinical and non-clinical events in a consistent manner. Consequently, it becomes easier and faster to carry out searches using the structured data format. Unstructured data in the form of clinical notes may also be used to capture additional information if the clinician finds the coded form inadequate. As a result, EHR data is usually very large, complex, heterogeneous, hierarchical, and varies in quality [120]. Electronic health records (EHR) provide access to longitudinal data that may offer a unique clinical asset that can be used to predict future outcomes or diagnoses, opening opportunities

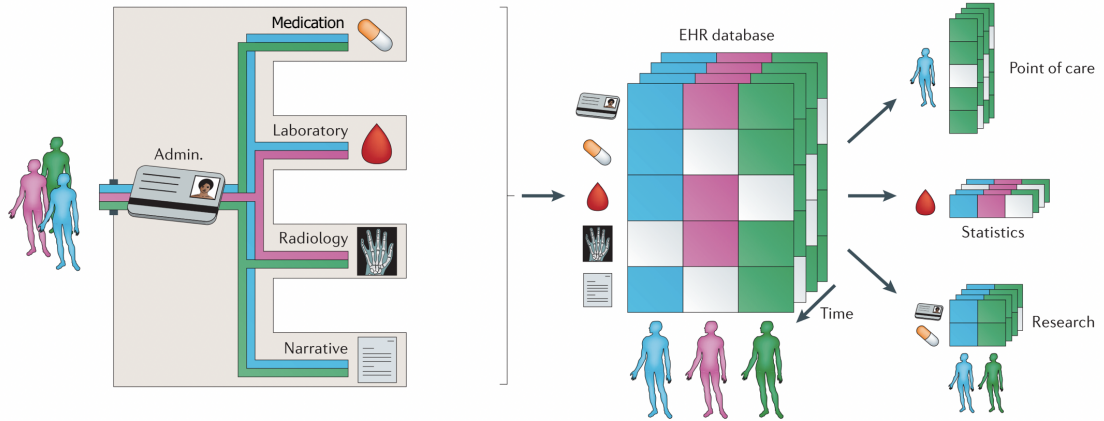


Figure 1.1: The electronic health record (EHR) of a patient can be viewed as a repository of information regarding his or her health status in a computer-readable form. An encounter with the health-care system generates various types of patient-linked data. In the example shown, a heterogeneous mixture of medication, laboratory, imaging and narrative data are all generated [126]

to personalise decision-making for a given patient [257]. EHR data is also used for audit, quality improvement, health service planning, epidemiological studies, and research [76]. Over time, the data grows rich in complex interactions, structures, and relations. It offers an opportunity to apply data mining techniques to harness patterns or trends for secondary healthcare use. Thus, epidemiologists and researchers are able to report interesting discoveries from studies conducted with high-volume, real-life clinical data. These studies apply statistical and multivariate analysis tools in order to report disease distributions, outcomes, prevalence, prognosis, establish risk associations to diseases and test the efficacy of a drug in real life. For instance, its use in replicating the results of randomised controlled trials (RCTs) [116].

### Problem of modelling with EHR

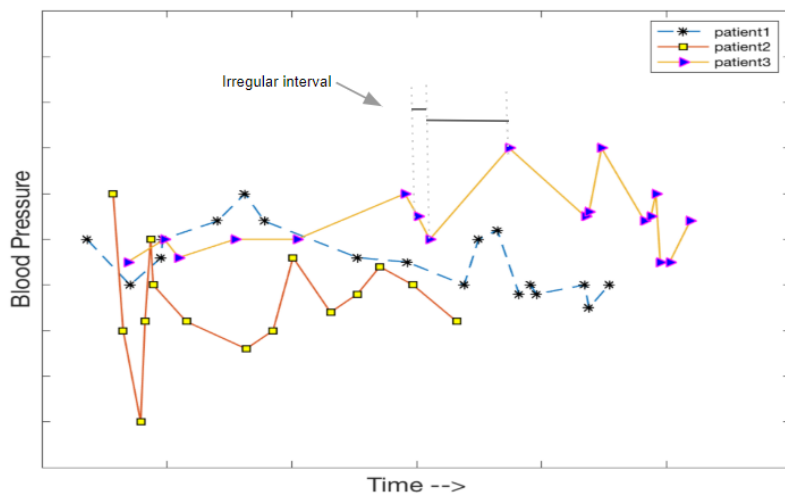



Figure 1.2: Example showing irregular intervals between longitudinal uneven-length blood pressure data for three patients

As more medical knowledge and discoveries accumulate, inferring knowledge using basic database management tools or traditional data processing applications becomes difficult [231]. The use of

routine EHR data for research can be challenging because the data were not originally collected for research purposes [116]. The data capture is typically driven by the purpose of the clinical encounter that is described by the data, and therefore other things may not be recorded [163]. Clinical encounters usually involve an interactive question-and-answer session between the clinician and patient. This could potentially lead to inconsistencies in the database because two clinicians may not follow the same line of inquiry. At the population level, the heterogeneous longitudinal data is made up of arbitrary-length and irregularly sampled entities. EHR can be described as sparse, incomplete with missing values, noisy, inconsistent, heterogeneous, inaccurate, and high-dimensional [120, 252]. A complete and accurate dataset is important in clinical research, as missing data are hard to interpret [76]. In addition to evidence-based requirements in healthcare research, a solution that is robust requires that we address these issues, specifically the irregular sampling of arbitrary length heterogeneous data. The examples displayed in Figure 1.2 show an uneven length and unequal intervals between recorded BP data for three patients. Goldstein et al. conducted a systematic review of clinical prediction studies using EHR data and found that only 58 out of the 107 studies reviewed assessed missingness [100]. Multiple imputation was seen in their study as the most common strategy for dealing with missing data. Prior to applying a data mining tool, the data may be transformed in ways that undermine their meaning, are unrecoverable for research, have unknown provenance, have insufficient granularity, and are incompatible with research protocols [116]. Another problem inherent to EHR data includes modelling repeated measurements [100]. Conducting research on problematic EHR data, therefore, has the likelihood of introducing bias into the outcome.



ID	Dob	Gender	Ethnicity	Deprivation Score	BMI Date	BMI	Systolic BP	Diastolic BP	Cholesterol Date	Cholesterol Value	Creatinine Date	Creatinine Value	eGFR Date	eGFR	SmokingDate	SmokingCat
1	2009-07-01	M	White	441	2013-10-24	14.6	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
2	1937-02-01	F	NA	23963	2014-07-10	24.3	130	70	2014-07-10	5.8	2014-07-10	64	2014-07-10	60	2014-07-10	Never smoked
3	1961-03-01	F	NA	16408	2013-10-03	29.7	146	91	2013-10-03	4.68	2004-01-05	73	NA	NA	2014-10-08	Active smoker
4	1962-03-01	F	White	10281	2012-10-08	38.4	140	82	2012-03-05	5.2	2012-03-05	110	2012-03-05	46	2014-09-30	Ex smoker

ID	Age	Gender	Ethnicity	Deprivation Score	BMI	Systolic BP	Diastolic BP	Cholesterol Value	Creatinine Value	eGFR	Smoking Cat
1	13	1	1	441	14.6	NA	NA	NA	NA	NA	NA
2	85	0	NA	23963	24.3	130	70	5.8	64	60	0
3	61	0	NA	16408	29.7	146	91	4.68	73	NA	1
4	60	0	1	10281	38.4	140	82	5.2	110	46	2

Figure 1.3: Flat-file tabular representation of EHR data. Each row represents a patient and each column contains a variable. This is further processed into features by discarding some variables.

Modelling with EHR data requires overcoming the difficulties of raw input data. Transforming the raw input data into a format compatible with a multivariate analytical tool is usually the first preprocessing step undertaken. A common approach transforms the variables of interest into tabular flat files in the form of one line per record and one column per variable (See Figure 1.3). Potential information is therefore discarded in favour of a simple representation. In addition, we are restricted in our choice of analytical tools that can work with this form of representation. A second option transforms the data into real-valued feature vectors. This form of representation allows us to take advantage of linear algorithms that are based on sound mathematical, statistical, and optimization techniques in order to gain insight into the data. This approach is appropriate for tackling numeric data problems. However, it is inadequate for dealing with non-numeric data or structural pattern recognition problems. It is unable to extract relations from unstructured data, nor can it be applied directly to model the spatial and temporal elements of the data simultaneously. By this, we mean dealing with categorical data (see a few examples displayed in Table 1.1) at the same time as numeric data (see a few examples displayed in Table 1.2), or both categorical and numeric data typical of

medication data (see a few examples displayed in Table 1.3). Besides, feature engineering is an expensive activity that requires expert knowledge.

On the other hand, representing non-linear heterogeneous data as a discrete sequence of symbols makes it possible to model interactions within variables and to model complex relationships that may exist. The classification performance of a speaker verification task was improved by adopting a sequence-based learning approach [157]. We can therefore use sequence alignment learning methods to overcome the problem of modelling unequal and non-aligned events that occur due to the irregular time intervals at which medical events are recorded. Doing so with the tabular approach or vectorial representations will be difficult to accomplish. Symbolic sequence learning has been used successfully in the analysis of optical character recognition problems, in biomedical studies such as the analysis of protein-protein interaction, biological sequence classification [141], speaker recognition [46], video concept detection [21], and DNA sequencing. Symbolic sequence learning is well suited for applying the kernel approach to tackling problems of pattern recognition.

Patient ID	Event date	Read Code	Read Term
ID6568	19960412	66A..00	Diabetic monitoring
ID6568	19960412	6872	Diabetes mellitus screen
ID6568	19960412	6781	Health education offered
ID6568	19960412	6673	Driving licence
ID6504	19960412	H05z.00	Upper respiratory infect.NOS
ID6280	19960412	H06z011	Chest infection
ID5587	19960412	14L..00	H/O: drug allergy
ID5060	19960412	9OW4.00	New patient screen 1st letter

Table 1.1: Sample view of categorical EHR data (see Appendix B for more examples)

### Multivariate analysis of EHR

Despite these known issues, clinical diagnosis or prognosis can be presented as supervised machine learning in the form of a classification or regression problem. It can also be presented as an unsupervised learning problem where exploratory techniques such as cluster analysis or anomaly detection can be applied. Evaluation of EHR data with these techniques may lead to the discovery of hidden knowledge within the data that could significantly enhance our understanding of disease progression and management [120]. Recent studies where these methods have been applied to biomedical problems include breast cancer diagnosis [267], early diagnosis of smoking-induced respiratory changes [10], cancer prognosis [140], predicting the outcome of clinically isolated syndrome [254], predicting outcomes in chronic kidney disease [26], predicting hospitalisation due to heart diseases [68] or in early prediction of magnetic resonance imaging (MRI) based Alzheimer’s conversion in mild cognitive impairment (MCI) subjects [170], to mention a few examples. Commonly used algorithms in big medical data analytics include Support Vector Machines (SVM), Logistic regression, Artificial Neural Networks (ANN), Random Forests, and Bayesian networks. Evidence from the literature shows that machine learning, to some extent, improved the accuracy of disease diagnosis and prognosis, enabled early detection of patients at risk, enabled personalised patient care, and allowed the choice of the best medication for patients. Wu et al. showed that it was possible to predict patients at risk of suffering chronic heart failure six months in advance [257]. While Amaral et al. hypothesised that machine learning models could be used to detect smoking-induced respiratory changes early when pathologic changes were still potentially reversible and, as a result, help prevent chronic obstructive pulmonary disease (COPD) [10]. Overall, the use of machine learning can aid in clinical decision-making.

No:	Event date	Read Code	Read Term	Value	Normal Range	
ID8329	20000525	423..00	Haemoglobin estimation	14.4	11.5	16
ID8329	20000525	425..00	Haematocrit - PCV	0.43	0.36	0.46
ID8329	20000525	426..00	Red blood cell (RBC) count	4.89	4	5.2
ID8329	20000525	428..00	Mean corpusc. haemoglobin(MCH)	29.4	25	35
ID8329	20000525	429..00	Mean corpusc. Hb. conc. (MCHC)	33.3	31	36
ID8329	20000525	42A..00	Mean corpuscular volume (MCV)	88.3	80	100
ID8329	20000525	42B6.00	Erythrocyte sedimentation rate	19	1	12
ID8329	20000525	42H..00	Total white cell count	8.46	4	10.5

Table 1.2: Sample view of numeric EHR data (see Appendix B for more examples)

## 1.2 Gaps Identified

A canonical framework for representation, analysis, and inference that is based on in-congruent, multi-source, and multi-scale biomedical data did not exist [83] at the onset of this research. Earlier studies [252] had highlighted that it was difficult to deal with modelling heterogeneous data, especially with missing values. Despite the widening application of several machine learning techniques to EHR, the problems of harnessing temporality, representing irregularly sampled data, and dealing with heterogeneity still persist. Particularly, the studies [183, 223] identified irregular sampling of data and varying lengths of available patient histories as problems with EHR data. The recent systematic reviews [218, 260] of several deep learning methods applied to EHR identified heterogeneous data as a major challenge to deal with. Their findings reveal that detecting patterns among multimodal data can increase the accuracy of diagnosis, prediction, and the overall performance of the learning system. Nevertheless, the majority of the studies [218] focused on the code-based representation of clinical concepts and patient encounters. They ignored many important, real-valued measurements associated with items such as laboratory tests, intravenous medication infusions, vital signs, and more. Accordingly, the need to address these diverse entities directly is expressed as an expectation of future research. As noted, “it appears that the next logical step for deep EHR research is the development of frameworks that utilise all types of patient data, not sets of homogenous data types considered in isolation.” Thus, deep learning research based on mixed data types is still ongoing and has the potential for huge benefits.

While techniques such as DTW may have been widely used by several communities to tackle time-domain problems, how best to account for the temporality of complex longitudinal clinical data remains an important research question [266].

Patient ID	Event date	Code	Name	Form	Str	Qty
ID5535	19880427	5271007	BISACODYL TAB 10	TAB	10	56
ID5562	19880427	49223020	ELTROXIN TAB 100	TAB	100	100
ID5562	19880427	51609020	GAVISCON TAB	TAB	0	180
ID5562	19880427	53819020	DELTACORTRIL ENTERIC TAB 2.5	TAB	2.5	100
ID5562	19880427	53647020	ZOVIRAX CRE 5	CRE	5	1
ID5562	19880427	49223020	ELTROXIN TAB 100	TAB	100	100
ID5562	19880427	53819020	DELTACORTRIL ENTERIC TAB 2.5	TAB	2.5	100
ID5562	19880427	49223020	ELTROXIN TAB 100	TAB	100	100

Table 1.3: Sample view of medication data with numeric and categorical content (see Appendix B for more examples), where **Str**: Strength and **Qty**: Quantity

---

## 1.3 Proposed Research

### 1.3.1 Research Question

Can the use of a supervised machine learning kernel framework improve the performance of predictive modelling of heterogeneous, uneven-length, and irregularly sampled primary healthcare data?

### 1.3.2 Aim of the Research

Apply kernel-based machine learning as an effective data-driven approach to predicting people at risk of developing type 2 diabetes.

### 1.3.3 Research Objectives

- Create bespoke variants of edit distance-based kernels that can model the various relations and interactions.
- Apply the edit kernels to address the problem of irregularly sampled arbitrary length data.
- Use a multi-kernel learning approach to address the problem of modelling heterogeneous entities with diverse physical characteristics.
- Establish the most appropriate kernel construction method to utilise the edit distance measure in developing kernels with discriminative capabilities.
- Search for the best predictive value by applying a multi-kernel learning (MKL) approach.
- Use the knowledge gained from the learning process to predict those at risk of developing type 2 diabetes from a prior episode of elevated BP.

### 1.3.4 Specific problem to be addressed

This research will specifically seek to address the problem of modelling with heterogeneous and irregularly sampled data of uneven length. It will develop bespoke kernel functions derived from variants of the edit distance proximity measure. It will apply a multi-kernel learning (MKL) technique to combine several base kernels generated from all available data types. It will achieve the stated objectives by developing an effective data-driven approach to predicting people at risk of developing type 2 diabetes. This research will examine the utility of the proposed framework as a potential prognosis tool with a series of experiments. The patient's behaviour, characteristics, medication, health trajectory, and status are extracted from various relational databases that make up the EHR database. These heterogeneous timestamped clinical and nonclinical events exist in categorical and numerical real-valued test measurements (see illustration in Figure 1.4). The experiments are designed to exploit heterogeneity within the data, specifically by incorporating both spatial and temporal information regarding patient behaviour, while at the same time overcoming the problem of irregularly sampled data of uneven length.

As a case study, this research seeks to address the question of how to characterise an elevation in blood pressure of 130/80 mm Hg in healthy patients as a warning for developing type 2 diabetes. Elevated BP is a modifiable risk factor that is also monitored in people at risk of developing hypertension. A key question is therefore whether susceptible patients with an occurrence of elevated BP prior to the onset of type 2 diabetes share similar behaviour? If such patterns exist within the data, can the approach proposed in this research serve as a preventive measure?

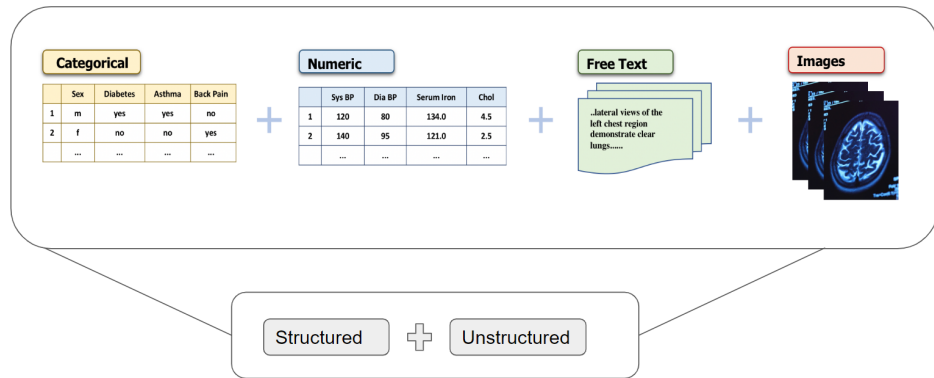


Figure 1.4: An illustration depicting entities with diverse physical characteristics that make up heterogeneous EHR data.

### 1.3.5 Rationale

This research builds on the recommended future work highlighted in the study [252]. It seeks to answer the question: “To what extent can we predict a selected patient’s disease diagnosis/prognosis from longitudinal primary care data using the kernel-based framework?” While several multivariate analytical works have been applied in addressing healthcare problems, modelling with routinely collected primary care data is still fraught with challenges such as irregularly sampled arbitrary length data made up of entities of diverse physical characteristics. These problems still persist and pose a challenge for standard multivariate analytical tools. The motivation for this work is to show that it is possible to use the kernel framework in conjunction with the edit distance to create bespoke edit kernel functions capable of addressing these challenges by retaining the data in its native form as a discrete sequence of symbols, while at the same time showing that it is possible to use the framework to exploit heterogeneity within the data. Specifically, by incorporating both spatial and temporal information regarding patient behaviour and, at the same time, overcoming the problem of irregularly sampled data of uneven length.

## 1.4 Contributions

We summarize the contributions to be made by this research below:

- Implement the kernel framework in addressing problem of heterogeneous data: The kernel framework provides a principled way of addressing heterogeneity that exists within EHR data from a primary care context. The use of MKL enables modelling with the full breadth of routinely collected data, irrespective of physical characteristics. This research applies MKL to address heterogeneity.
- Use elastic edit distance to address the problem of irregularly sampled data of arbitrary length. By this approach we apply the data in its native form as a discrete sequence of symbols; thus addressing the challenge of finding inappropriate representation of data with strenuous and expensive feature extraction. Information that would otherwise be lost with traditional vector space models, such as the spatial and temporal aspects of the data, is incorporated into the model.
- Use the framework to develop a disease prognosis tool. We leverage the utility of the framework in overcoming the identified challenges in developing a disease prognosis tool. Specifically, address the problem of identifying susceptible healthy patients likely to succumb to type 2 diabetes given a prior episode of elevated blood pressure of 130/80 mmHg. Consequently, we show this



---

framework is suitable as a tool that can aid in clinical decisions and in the management of a chronic disease by extracting actionable insight from problematic primary care data.

## 1.5 Expected Impact

It is expected that the outcome of this research will contribute to the discussion on the relative strengths and weaknesses of modelling options using EHR, on kernel methods with an emphasis on edit distance-based kernels, multiple kernel learning, and optimization techniques. It will show that it is possible to use machine learning to extract useful knowledge from noisy, incomplete, heterogeneous, irregularly sampled, and arbitrary-length EHR data by utilising the proposed kernel-based framework. It will provide a means of implementing a prognosis tool that can be incorporated into GP IT systems within the primary care setting. Such a tool can help as a preventive measure in the early identification and targeting of healthy patients susceptible to type 2 diabetes with appropriate lifestyle interventions during routine care.

## 1.6 Proposed Kernel Framework

The kernel framework applies to the implicit mapping of non-linear data points into an embedded high-dimensional feature space where, for instance, linear separation can take place. It is computationally efficient because we do not need to explicitly compute the coordinates of the mappings in the embedded space; rather, we use a pairwise similarity function (also called a kernel function), applied to the raw input data. The kernel function corresponds to an inner product between each pair of points in the embedded feature space. This is known as the “kernel trick” and is applicable to both statistical and syntactic data structures. A valid kernel function obeys Mercer’s conditions. It is thus symmetric and positive semi-definite (PSD). Multiple kernels can be constructed for each heterogeneous variable and combined algebraically using a suitable multi-kernel learning (MKL) technique. Linear algorithms like SVMs or Gaussian processes that are based on the inner products of the input space can then be used to find linear separability in the feature space. The kernel approach is modular because we can separate the construction of kernels from the choice of linear algorithms. Evaluating the kernel function on all data points yields a kernel matrix that encodes the relative positions of the data points. The optimization process to find the separating hyperplane in a kernelized discriminative model is independent of the dimensionality of the raw input data. This gives the model an advantage in overcoming the problems of high-dimensional data. The neutral point substitution approach [253] used in dealing with missing values can also be applied to the kernel method. The kernel model allows domain knowledge to be factored into the development of kernels, thereby increasing the range of problems that can be tackled with the kernel approach. Sequence kernels developed from symbolic structures allow us to incorporate both spatial and temporal elements into the model. Variants of base sequence kernels with elastic similarity measures such as the edit distance and dynamic time warp (DTW) can be used effectively in predictive modelling of EHR data.

The kernel framework provides a powerful and robust tool for tackling the various issues highlighted with the primary healthcare data. This research will therefore focus on representing the data as a discrete sequence of symbols and deriving variants of the edit distance-based kernels for use in prediction modelling with primary healthcare data. It will seek to answer the question: “To what extent can we predict a selected patient’s disease diagnosis or prognosis from heterogeneous longitudinal primary care data using the edit distance based sequence kernels?” This research will also seek to show that predictive modelling of EHR data can produce actionable insights from the data.

---

## 1.7 Advantages of the Kernel Framework

This section presents some of the advantages of the kernel framework, which include overcoming the problems of modelling irregularly sampled, heterogeneous, and longitudinal EHRs.

- The kernel matrix provides all the information that is required by the learning algorithm. Its dimension depends more on the number of data examples than on the number of variables. Therefore, the computational feasibility of finding the solution to the optimization objective of the learning algorithm can easily be achieved when we have high-dimensional data with fewer data examples. We can potentially utilize the full complement of the predictor variables.
- The kernel matrix as the only input into a kernel-based learner ensures minimal contact with raw EHR data, thus making the kernel approach well suited for handling sensitive clinical information.
- Multi-kernel learning (MKL) makes it possible to develop and combine kernels from all predictor variables present, irrespective of the raw data format or characteristics thus, providing a solution to the problem of modelling with heterogeneous data and avoiding expensive feature extraction processes.
- The kernel method can be used to effectively tackle non-linear problems. It implicitly transforms non-linear input data into a high-dimensional feature space where linear patterns can be found with stable linear algorithms with well-understood statistical properties.
- The user-definable kernel function allows us to incorporate domain knowledge into its development. We can therefore tackle a wider range of problems.
- Kernel functions are reusable and can be tailored to suit specific tasks instead of applying a generic one-size-fits-all approach to all problems.
- The kernel method is computationally efficient because we can take advantage of its strengths without explicitly computing the transformation into the high-dimensional feature space. Rather, we simply focus on hand-crafting kernel functions in the raw input space.

## 1.8 Published Material

The published collaborative paper [184] titled **A novel kernel-based approach to arbitrary length symbolic data with application to type 2 diabetes risk** applied traditional edit distance in addition to 3 normalization techniques described in the Methodology section 4. These proximity measures were implemented as kernel functions using the distance substitution method and applied to symbolic EHR data. The paper successfully addressed the problem of irregularly sampled arbitrary length EHR and showed the discriminative capabilities of the model improved by combining poor standalone kernels into a single model by utilising MKL. It essentially proposed the featureless edit kernel strategy as a generally preferable form of EHR-based machine learning on the basis of its implicit retention of all clinically relevant information that may otherwise be lost in the feature representation process.

## 1.9 Type 2 Diabetes Mellitus

Type 2 diabetes is the case to be addressed in this research; therefore, this section gives a brief background regarding the chronic disease. Type 2 diabetes mellitus (T2DM) is a chronic metabolic disorder in which optimal glycaemic control tailored to the individual is recommended in order to

---

reduce the risk of long term complications. Lifestyle interventions such as regular exercise, weight loss, and a healthy diet are advised on diagnosis. Pharmacotherapy with a single non-insulin oral hypoglycemic agent (OHA) can be prescribed at the same time or shortly thereafter. Metformin of the biguanide drug class is generally the glucose lowering agent of first choice in those newly diagnosed with T2DM. It is prescribed in conjunction with suitable lifestyle advice and the management of cardiovascular risk factors. It acts by suppressing elevated gluconeogenesis and improving insulin sensitivity in the liver and muscles. Its mode of action rarely leads to the hypoglycaemia seen with some other glucose-lowering agents. In addition, it tends to be weight neutral and can result in weight loss.

The antihyperglycaemic efficacy of metformin increases with increasing daily doses between 500 mg and the upper limits of the recommended daily dosage ( $\geq 2000$  mg/day) [214]. Dose titration is according to the HbA1c response and patient tolerance. Metformin causes mild gastrointestinal (GI) side effects like diarrhoea, nausea, a metallic taste, and abdominal pain. These side effects were seen in 28% [97] and in less than 20% [121] of the referenced study population. While NICE guidelines indicate its use is contraindicated or not tolerated in approximately 15% individuals [173]. Studies have shown that persistence rate estimates after 12 months for people with T2DM prescribed with metformin varies from 30% [222], 60% [38], to 65% [107]. The GI side effects could also lead to lower physical and mental Health Related Quality of Life (HRQoL) and may result in low drug utilisation or physician reluctance to optimally titrate the metformin dose [93]. Switching to once-daily extended-release metformin MR presents an effective and well-tolerated therapeutic option for delivering metformin in a convenient manner [95]. Increased drug utilisation, improved glycaemic control and hence decreased risk of the associated micro and macrovascular complications were reported in these studies [9, 91, 125]. According to the study by Blonde et al., [36] GI indications occurred less in the 205 study population after switching from metformin standard-release to metformin MR. The frequency of GI side effects was 26.34% vs. 11.71% (after switching) ( $p = 0.0006$ ), and the frequency of diarrhoea was 18.05% vs. 8.29% (after switching) ( $p = 0.0084$ ).

Metformin is contraindicated by congestive heart failure, renal impairment due to the risk of lactic acidosis and advanced age, over 80 years. Its cardiovascular benefit in terms of reduced cardiovascular-related deaths was demonstrated by the UK Prospective Diabetes Study (UKPDS), particularly among newly diagnosed people with T2DM who had BMIs in excess of 30 among the 342 patients allocated metformin in the study. Metformin as the first line monotherapy of choice [173] is supported by systematic reviews and meta-analysis [32, 164, 210]. It may also have other incidental benefits, such as acting as a geroprotective agent [47], prevent people with impaired glucose tolerance (IGT) from progressing to T2DM [138], improve fertility [122] lower the risk of a decline in eGFR, ESRD or death regarding kidney function [123], suppress the risk of certain cancers and improve chances of survival [65, 88, 102]. The anti-inflammatory property of the drug may have a role in the treatment of rheumatoid arthritis [171]. Women with polycystic ovarian syndrome (PCOS) and T2DM fared better than non-users [172]. It is also found to be a safe insulin sparing agent [168, 241]. Many clinicians use it in type 1 diabetes, where patients are obese and have inherent insulin resistance.

Following the diagnosis of T2DM, therapeutic regimens are intensified if the blood glucose level remains high or complications arise due to the presence of comorbidities. A second OHA is combined with metformin, and when the glycaemic target is not met, a third OHA may be added. Common co-therapies that are prescribed with metformin include sulfonylurea, a DPP-4 inhibitor, or SGLT2 inhibitors [173]. Other classes of drugs available include, but are not limited to, meglitinides, thiazolidinediones (TZDs), glucagon-like peptide-1 (GLP-1) analogues, and insulins. Their formulations, when combined with metformin, can alter the interactions in users, leading to

---

some side effects like hypoglycaemia and weight gain. The patient’s individual circumstances and drug tolerance, among other factors, are considered when choosing additional therapy. Given the benefits of metformin, patients are encouraged to persist with the prescribed medication until discontinuation. While this may be desired, they often fail to stick to the therapeutic regimens as directed, even when the GI effects of metformin are tolerated.

Reasons why patients who are tolerant of metformin fail to persist with the medication vary. They range from patient-related, socioeconomic, condition-related, health systems, and therapy-related reasons. Certain factors may exist as hidden variables that are harder to observe. In some studies, gender, age [230], pre-existing therapy [201], adverse effect [255], switching medication [58] dose [45], presence of comorbidities [58], multi therapy regimen [198], and lack of perceived need [255] were identified factors that can influence persistence outcomes. The development of renal disease will also lead to discontinuation. Clinicians are advised to stop metformin if the eGFR is below  $30 \text{ mL/min}/1.73\text{m}^2$  or reduce the dosage if the eGFR falls below  $45 \text{ mL/min}/1.73\text{m}^2$  [173]. Tackling therapy-related factors can lead to an informed choice of alternative medication. However, the same approach will become ineffective when dealing with non-therapy-related factors. Patients who do not persist with metformin can experience worse clinical and economic outcomes, including poorer haemoglobin A1c (HbA1c) control, greater hospitalization rates, higher mortality rates, and higher healthcare costs [45]. Understanding persistence is important because it allows clinicians to understand factors associated with a lack of persistence and to efficiently identify patients for individualised intervention [11]. It will also aid in therapeutic decision-making. The clinical and economic outcomes related to low persistence can also be evaluated. Cox proportional hazards, Kaplan-Meier survival curves, and logistic regression are examples of multivariate approaches commonly used in analysing medication persistence rates.

Studies have reported that current methods of treating the disease are both uncertain and costly, and so prevention becomes an important step towards reducing the burden of care [138, 179]. Thus, it has become a clinical imperative to explore predictive models based on EHR data capable of identifying those most susceptible to developing the disease, given that evidence of impending lifestyle choices can be gleaned from various clinical entities holding historic details about the patients. For instance, elevated blood pressure (BP) measurements constitute one of the key modifiable risk factors seen in people at high risk of diabetes [7] and may help inform intervention via early education on lifestyle choices.

Prognosis tools that carry out risk assessment, such as QDiabetes [2], FINnish Diabetes RIsk SCore (FINDRISC) [1], and the “Know Your Risk” tool from Diabetes UK [80] are currently available online. FINDRISC is commonly used in Europe [105]. Although these tools are accessible to patients and present measures for indicating likelihood of the disease, they are unlikely to catch all susceptible patients being based on limited data (it has been found empirically that several conditions associated with increased risk of diabetes are not fully captured by Qdiabetes [118]). Thus, while these simpler models are easier to implement, they may oversimplify complex relationships that include large numbers of risk factors with non-linear interactions [132]. In this context, UK NHS Nice guidelines on preventing type 2 diabetes recommend, where possible, computer-based risk-assessment tools using available routine EHRs [174]. This is backed by evidence from studies [6, 194] indicating machine learning prognostic models developed from EHR data usually perform better than simple statistical prognostic models.

Several works have applied machine learning algorithms in identifying people at risk of developing type 2 diabetes. Recent examples include the ensemble-based approach of [8, 179], the Multi-Layer

---

Perceptron (MLP), AdaBoost (AD), Trees Random Forest (TRF), the Hidden Markov Model (HMM) of [199], Support Vector Machine (SVM), and the Gradient Tree Boosting (GTB) approach of [158]. A previous review paper [25] however highlighted a widespread problem of poor methodologies in developed risk tools and also the issue of inconstant use of data and predictor variables (for instance, 12 predictor variables are used in [167] compared to 1312 predictor variables in [179]). The UK National Screening Committee report [247], however, indicates that while a small set of risk indicators can have its advantages since they may easily be extracted from EHR data, they are less likely to include valuable information such as waist measurement and a linked family history of the disease that are strong indicators for determining the level of risk of certain patients.

This research is focused on addressing the problem of developing an effective data-driven approach to predicting people at risk of developing type 2 diabetes. It seeks to address the question of how to characterize an elevation in blood pressure of 130/80 mm Hg in healthy patients as a warning for developing type 2 diabetes. Elevated BP is a modifiable risk factor that is also monitored in people at risk of developing hypertension. A key question is therefore whether healthy patients with an occurrence of elevated BP prior to the onset of type 2 diabetes share similar behaviour? If such patterns exist within the data, can the approach adopted in this study serve as a preventive measure?

## 1.10 Dissertation Layout

The rest of this document is structured as follows. [Chapter 2](#) presents the theoretical underpinning of the kernel framework. [Chapter 3](#) details a comprehensive literature review that gives a historic overview of the development and adoption of technology in primary healthcare delivery. The problems faced with the secondary use of the accumulated data pertaining to multivariate analysis are stated. In addition, elastic proximity measures with a detailed overview of edit distance proximity measures and their use with the kernel framework are introduced, and lastly, an overview of type 2 diabetes mellitus is documented. [Chapter 4](#) covers the methodology adopted for the research in detail. This includes describing variants of edit distance proximity measure and kernel construction methods. [Chapter 5](#) presents the experimental objectives, findings, and results. [Chapter 6](#) presents a detailed discussion of the findings presented in the previous chapter. It states the future research, its limitations, and conclusion.

## Chapter 2

# Proposed Kernel Framework

In this chapter, we introduce some of the theoretical underpinnings of the proposed kernel framework.

### 2.1 Linear Functions

The appeal of the kernel framework stems from its suitability for addressing problems of a non-linear nature using linear functions in a feature space created by a non-linear feature map. The linear functions are preferred because they are interpretable, robust, and have mature statistical properties. A linear function is a map  $f$  between vector spaces. It defines a linear relationship between two or more vector spaces and has the following properties:

$$f(\mathbf{x} + \mathbf{y}) = f(\mathbf{x}) + f(\mathbf{y})$$

$$f(\alpha\mathbf{x}) = \alpha f(\mathbf{x})$$

where  $\mathbf{x} \in X$  and  $\mathbf{y} \in Y$  are elements of vector spaces and  $\alpha \in \mathbb{R}$ .

### 2.2 The Learning Problem

The goal of supervised learning is to find the unknown target functional mapping of  $X \mapsto Y$  from a data  $(\mathbf{x}, Y)$  where  $x \in X$  and  $Y \in \{1, -1\}$  for a binary classification problem by minimizing some error function. Then use the knowledge gained to generalise on previously unseen data. This goal is achieved with limited sample-sized data and is based on the assumption that the entire data set, including the unseen examples, is generated from the same process, i.e., the same probability distribution.

### 2.3 Hyperplane Classifiers

A hyperplane classification model can be constructed with a real-valued function  $f : \mathbf{X} \subseteq \mathbb{R}^n \mapsto \mathbb{R}$ . We can build a classifier using the following affine function;

$$f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle = \sum_{i=1} w_i x_i + b$$

where  $b$  is the offset parameter to be learned from the data. It controls where the hyperplane lies geometrically along the  $x$ -axis of the Euclidean plane. The hyperplane lies at the origin if  $b$  is set to zero.  $\mathbf{w}$  is a weight parameter that is applied to the features  $\mathbf{x} \in X$ . The weighting  $\mathbf{w}$  quantifies the relationships with other objects and allows us to apply a threshold to classify objects of a certain size

in a similar fashion. We can apply a threshold to function  $f(\mathbf{x})$ , such that the input  $\mathbf{x} = (x_1, \dots, x_n)$  is assigned the positive class +1, if  $f(\mathbf{x}) \geq 0$  and otherwise to the negative class -1.

This is essentially the objective of learning patterns from data. As a result, we can use the concept of distance between objects in the feature space  $\|x_i - x_j\|$  to classify objects. If the distance between objects is small, the distance between the resulting real-values will also be small. The weight vector is usually unknown but assumed to be of unit length. It is perpendicular to the hyperplane separating the points in the feature space. Geometrically, new points are projected onto the weight vector, and the side they fall on within the hyperplane determines the class to which they belong.

Functionally, a classifier is thus built with the decision function:

$$f_w(\mathbf{x}) = \text{sign}(\langle \mathbf{x}, \mathbf{w} \rangle + b)$$

There is usually the need to recode the features by using a non-linear mapping function  $\phi = (\phi(x_1), \dots, \phi(x_n))$ . The new representation,  $\phi(\mathbf{x})$ , which can also be of higher-dimension, can make it easier to find patterns in the feature space because it can also encode other information such as correlations between variables. This then gives the decision function as:

$$f(\phi(\mathbf{x})) = \text{sign}(\langle \phi(\mathbf{x}), \mathbf{w} \rangle + b)$$

The beauty of this linear function is the fact that the weight parameter vector  $w$  lies within the linear span of the data points  $x$ . It can be represented as the linear sum of all the training data points.

$$w = \sum_{i=1}^m \alpha_i \phi(x_i)$$

Replacing  $w$  in the linear function above, we have

$$f(\mathbf{x}) = \left\langle \sum_{i=1}^m \alpha_i \phi(x_i), \phi(x) \right\rangle$$

The problem to solve becomes one of finding  $w$  or finding  $\alpha$ . Nevertheless, this introduces us to a solution that is based on the dot product  $k(\mathbf{x}, \mathbf{x}) = \langle \phi(\mathbf{x}), \phi(\mathbf{x}) \rangle$ . Any learning algorithm based on taking the dot product of the data points can thus be substituted with a kernel and known as a kernel classifier.

## 2.4 Kernel Definition

Having laid the foundation, it is necessary to give a formal definition of a kernel function. Via the so-called kernel trick, a kernel is equivalent to an implicit mapping of entity pairs into a high-dimensional feature space, followed by a vector product in that space. It is thus a symmetric function  $K : X \times X \mapsto \mathbb{R}$  such that,

$$\forall x_i, x_j \in X, \quad k(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle \quad (2.1)$$

where  $\phi : X \mapsto F$  is a function map  $\phi$  that transforms the input  $X$  into a high dimensional feature space  $F$ . A valid kernel function is positive definite if it satisfies the condition

$$k(x_i, x_j) = \sum_{i,j=1}^n c_i c_j k(x_i, x_j) \geq 0 \quad (2.2)$$

for any  $x_1, \dots, x_n \in X$  and  $c_1, \dots, c_n \in \mathbb{R}$  or, equivalently, that all eigenvalues of its matrix are non-negative. See Figure 4.2 for the conceptual framework.

### 2.4.1 Positive definiteness

A kernel function is positive definite if the equation 2.2 holds. The resultant kernel matrix produced by evaluating the kernel function  $K$  on data  $X$  yields non-zero eigenvalues. The kernel matrix is said to be positive semi definite (PSD), if

$$\mathbf{c}'\mathbf{K}\mathbf{c} \geq 0$$

for all vectors  $\mathbf{c}$

This can be shown as follows: kernel matrices are PSD due to non negative norms.

$$\begin{aligned} \mathbf{c}'\mathbf{K}\mathbf{c} &= \sum_{i,j=1}^m c_i c_j K(x_i, x_j) = \sum_{i,j=1}^m c_i c_j \langle \phi(x_i), \phi(x_j) \rangle \\ &= \left\langle \sum_{i,j=1}^m c_i \phi(x_i), \sum_{i,j=1}^m c_j \phi(x_j) \right\rangle \\ &= \left\| \sum_{i,j=1}^m c_i \phi(x_i) \right\|^2 \geq 0, \end{aligned}$$

### 2.4.2 Inner product space - (Hilbert Space)

The induced inner product space is called a Hilbert space  $H$  if it has additional properties (separable and complete) that make it isomorphic to Euclidean space. This is necessary because we can take advantage of its geometrical properties in a search for the linear separability of the pattern. If  $H$  is a vector space over  $\mathbb{R}$ , then an inner product can be defined as a function

$$\langle \cdot, \cdot \rangle_H : \mathbf{H} \times \mathbf{H} \mapsto \mathbb{R}$$

on  $H$  if the following properties hold;

1. It is closed under linear addition and scalar multiplication;

$$\langle \alpha_i x_i + \dots + \alpha_n x_n, z \rangle = \langle \alpha_i x_i, z \rangle + \dots + \langle \alpha_n x_n, z \rangle$$

for all  $x, z \in X$  and  $\alpha \in \mathbb{R}$

2. It is symmetric:

$$\langle x, y \rangle = \langle y, x \rangle$$

3. It satisfies

$$\langle x, x \rangle \geq 0$$

and

$$\langle x, x \rangle = 0 \iff x = 0$$

4. We can thus define a norm of the space  $X$  by

$$\|x\|_2 = \sqrt{\langle x, x \rangle}$$

This has the following relations between the norm and the inner product:



- $|\langle f, g \rangle| \leq \|f\| \cdot \|g\|$  - Cauchy-Schwarz inequality
- $\|f + g\|^2 + \|f - g\|^2 = 2\|f\|^2 + 2\|g\|^2$  - parallelogram law
- $\langle f, g \rangle = \frac{\|f + g\|^2 - \|f - g\|^2}{4}$  - Polarization identity

Every Hilbert space is a Banach space, but the reverse is not true.

5. It is complete. A metric space  $(X, d)$  is called complete if every Cauchy sequence  $(x_n)$  in  $X$  converges to some point of  $X$ , where a Cauchy sequence  $X_n$  of  $X$  can be defined for every  $\epsilon > 0$  there exists  $N \in \mathbb{N}$  such that the distance  $\|x_n - x_m\| < \epsilon$  for all  $n, m \geq N$
6. It is separable. A space  $H$  is separable if for any  $\epsilon > 0$  there is a finite set of elements  $x_1, \dots, x_n$  of  $H$  such that for all  $x \in H$

$$\min_i \|x_i - x\| < \epsilon$$

In addition, we can also define an inner product on a space of functions  $F$ . If  $F = L_2(X)$  is a vector space of square integrable functions on a compact subset  $X$  of  $\mathbb{R}^n$  represented as

$$L_2(X) = \left\{ f : \int_X f(x)^2 dx < \infty \right\} \quad (2.3)$$

$\forall f, g \in X$ , we can define an inner product as

$$\langle f, g \rangle = \int_X f(x)g(x)dx \quad (2.4)$$

In order to apply the kernel technique in addressing a classification task, we first need to embed the input data using the non-linear feature map  $\phi$  then compute the dot product in the feature space. This is an onerous task that is computationally infeasible; thus less desirous. It is impractical to establish a suitable mapping function  $\phi$  that would yield higher dimensional features and then compute the inner products of the features. Therefore, an alternative approach is sought. Any symmetric pairwise similarity function  $k : X \times X \mapsto \mathbb{R}$  in the input space that satisfies Mercer's conditions is a valid kernel.

### 2.4.3 Mercer's Theorem

Mercer's theorem is achieved by studying the eigenvalue problem associated with the integral equations of the form.

$$\int_x k(x, x')f(x')dx' \quad (2.5)$$

This is the essence of the so-called \*kernel trick\* as we can bypass the feature map aspect and access the feature space for valid kernel functions. It defines the feature space in terms of an explicit feature vector, which differs from the function space used in the Reproducing Kernel Hilbert Space (RKHS) construction.

**Mercer's Theorem:** Let  $X$  be a closed subset of  $\mathbb{R}$ . Suppose  $K$  is a continuous symmetric function and the induced integral operator  $T_k : L_2(X) \mapsto L_2(X)$

$$(T_K f)(\cdot) = \int_X K(x, x')f(x')dx, \quad (2.6)$$

We say  $K$  satisfies Mercer's condition iff

$$\int_{X \times X} K(x, x')f(x)f(x')dx dx' \geq 0, \quad (2.7)$$

for all  $f \in L_2(X)$ . If we let  $\psi_i \in L_2(X)$  represent the eigenfunction of  $T_k$  associated with the eigenvalue  $\lambda_i \geq 0$  and  $\sum_{i=1}^{\infty} \lambda_i^2 < \infty$ . If this eigenfunction is normalized such that  $\|\psi_i\|_2 = \int_X \psi_i^2(x) dx = 1$ , i.e.,

$$\forall X \in X : \int_X k(x, x') \psi_i(x') dx' = \lambda_i \psi_i(x) \quad (2.8)$$

Then  $k$  can be expanded in a uniformly convergent series,

$$k(x, x') = \sum_{i=1}^{\infty} \lambda_i \psi_i(x) \psi_i(x') \quad (2.9)$$

which holds for all  $x, x' \in X$ .

The Mercer's theorem gives us a way of obtaining the features  $\phi_i$  from a given kernel  $k$ . Consider the map  $\phi$  from  $X$  into  $l_2$

$$\phi(x) = \left( \sqrt{\lambda_1} \psi_1(x), \sqrt{\lambda_2} \psi_2(x), \dots \right)^T \quad (2.10)$$

By the equation in 2.9 we have for each  $x, x' \in X$

$$k(x, x') = \sum_{i=1}^{\infty} \lambda_i \psi_i(x) \psi_i(x') = k(x, x') = \sum_{i=1}^{\infty} \phi_i(x) \phi_i(x') = \langle \phi(x), \phi(x') \rangle$$

The features  $\psi_i$  are called Mercer features and the mapping  $\psi_1(x), \psi_2(x), \dots$  is known as the Mercer map.

This gives us an easier way of learning with the kernel framework by choosing a Mercer kernel  $k$  that implicitly corresponds to the mapping  $\phi$ .

#### 2.4.4 Reproducing Kernel Hilbert Space (RKHS)

Let  $\mathcal{H}$  be the space of functions  $f$  that maps from a non-empty set  $\mathcal{X}$  to  $\mathbb{R}$

$$f : \mathcal{X} \mapsto \mathbb{R}$$

For a fixed  $x \in \mathcal{X}$  the linear map  $\delta_x : \mathcal{H} \mapsto \mathbb{R}, \delta_x : f \mapsto f(x)$  is called the (Dirac) evaluation functional at  $x$ . We have a Reproducing kernel Hilbert space (RKHS) if  $\delta_x$  is continuous  $\forall x \in \mathcal{X}$ . We can also write a canonical feature map  $\phi$  for every point in the feature space as  $\phi(x) = k(\cdot, x)$  i.e.

$$\forall x \in \mathcal{X}, k(\cdot, x) \in H$$

Consider any two functions  $f \in \mathcal{H}$  and  $g \in \mathcal{H}$  from this space that takes a linear combination of the function points i.e.  $f(\cdot) = \sum_{i=1}^n \alpha_i k(x_i, \cdot)$  and  $g(\cdot) = \sum_{j=1}^m \beta_j k(x_j, \cdot)$  to form vector spaces where  $\alpha \in \mathbb{R}^n, \beta \in \mathbb{R}^m$  and  $x_i, x_j \in \mathcal{X}$ . If we then take a dot product between  $f$  and  $g$

$$\langle f, g \rangle = \sum_{i=1}^n \sum_{j=1}^m \alpha_i \beta_j k(x_i, x_j) = \sum_{j=1}^m \beta_j f(x_j) = \sum_{i=1}^n \alpha_i g(x_i)$$

The result from taking a dot product of the functions gives the original definitions of the functions above. To illustrate the reproducing property, we can express the formulation above better by taking a function  $g = k(x, \cdot)$

$$\langle f, k(x, \cdot) \rangle = \sum_{i=1}^n \alpha_i k(x_i, x) = f(x) \quad (2.11)$$

Any kernel function that satisfies the positive semi-definite property induces a *Reproducing Kernel Hilbert Space (RKHS)*. We can show proof of this property:

$$\begin{aligned}
 \sum_{i,j=1}^m \alpha_i \alpha_j K(x_i, x_j) &= \sum_{i,j=1}^m \alpha_i \alpha_j \langle k(x_i, \cdot) \rangle_{\mathcal{F}} \\
 &= \left\langle \sum_{i,j=1}^m \alpha_i k(x_i, \cdot), \sum_{i,j=1}^m \alpha_j k(x_j, \cdot) \right\rangle \\
 &= \left\| \sum_{i,j=1}^m \alpha_i k(x_i, \cdot) \right\|^2 \geq 0,
 \end{aligned}$$

### 2.4.5 Cauchy-Schwarz Inequality for kernels

Kernels share this similar property as dot product. If  $k$  is a PSD kernel and  $x, x' \in X$ , then

$$|k(x_1, x'_2)|^2 \leq k(x_1, x'_1) \cdot k(x_2, x'_2) \quad (2.12)$$

### 2.4.6 Metric Spaces

Given a non empty set  $\mathcal{X}$  with the distance function  $d$  on  $\mathcal{X}$ , such that

$$d : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$$

A metric space is the ordered pair  $(\mathcal{X}, d)$  if for any  $x, x', x'' \in \mathcal{X}$  the following four axioms holds;

1.  $d(x, x') \geq 0$                       Non-negativity
2.  $d(x, x') = 0 \implies x = x'$       Reflective
3.  $d(x, x') = d(x', x)$               Symmetry
4.  $d(x, x'') \leq d(x'', x) + d(x'', x')$     Triangular inequality

The edit distance used in this study is a metric as it satisfies all four stated axioms.

### 2.4.7 Conditionally positive semi definiteness

Since PSD kernels are generalizations of vector products in the induced Mercer feature space, we can extend the concept of PSD kernels to a larger class of kernels known as conditionally positive kernels (cpd) expressed in terms of the norms of the embedding feature space. Thus, the norm  $\|\phi(x_i) - \phi(x_j)\|^2$  quantifying how close objects are in the feature space can be expressed in terms of the kernel function:

$$\|\phi(x_i) - \phi(x_j)\|^2 = k(x_i, x_i) + k(x_j, x_j) - 2k(x_i, x_j) \quad (2.13)$$

where  $k(\cdot, \cdot)$  is a kernel function

As a result, we are able to apply distance metrics, in this case edit distance, in the construction of kernels. A distance measure is said to be isometric to the L2-norm if the data can be embedded in a Hilbert space such that  $d(x, x_0) = \|\phi(x) - \phi(x_0)\|$  (this approach is termed a ‘distance substitution kernel’).

This intuition stems from norms being invariant to translations,  $x \mapsto x_i - x_0$  in contrast to dot products. The dot product of the translation can be expressed as

$$\langle (x_i - x_0), (x_j - x_0) \rangle = \frac{1}{2}(-\|x_i - x_j\|^2 + \|x_i - x_0\|^2 + \|x_0 - x_j\|^2) \quad (2.14)$$

For any  $x_0 \in X$  we show this to be a valid PSD kernel by

$$\sum_{i,j} c_i c_j \langle (x_i - x_0), (x_j - x_0) \rangle = \sum_{i,j} c_i \|(x_i - x_0)\|^2 \geq 0 \quad (2.15)$$

A conditionally positive definite symmetric  $n \times n$  matrix  $\mathbf{K}$  ( $m \geq 2$ ), on the other hand, also satisfies the condition in equation 2.2 for any  $x, \dots, x_n \in X$  and  $c, \dots, c_n \in \mathbb{R}$  but with the additional property

$$\sum_{i=1}^m c_i = 0 \quad (2.16)$$

## 2.4.8 Indefinite kernels

This is a class of kernels (see section 3.4.3) that violates the Mercer's condition of positive definiteness. Certain similarity measures that can adequately represent the problem domain often fall into this category of incorporating domain knowledge in developing kernel functions means.

## 2.4.9 Krein Space

An inner product space  $(\mathcal{K}, \langle \cdot, \cdot \rangle_{\mathcal{K}})$  is a Krein space if there exists two Hilbert spaces  $\mathcal{H}_+$ ,  $\mathcal{H}_-$  such that, all  $f \in \mathcal{K}$  can be decomposed into  $f = f_+ + f_-$ , where  $f_+ \in \mathcal{H}_+$  and  $f_- \in \mathcal{H}_-$ .

$$\forall f, g \in \mathcal{K}, \langle f, g \rangle_{\mathcal{K}} = \langle f_+, g_+ \rangle_{\mathcal{H}_+} - \langle f_-, g_- \rangle_{\mathcal{H}_-} \quad (2.17)$$

There is an associated Hilbert, where the difference in the dot products is replaced by a sum. We define the associated Hilbert space by decomposing a Krein space  $\mathcal{K}$  into Hilbert spaces  $\mathcal{H}_+$  and  $\mathcal{H}_-$ . Then we denote by  $\bar{\mathcal{K}}$  the associated Hilbert space defined by  $\bar{\mathcal{K}} = \mathcal{H}_+ \oplus \mathcal{H}_-$ , hence,

$$\langle f, g \rangle_{\bar{\mathcal{K}}} = \langle f_+, g_+ \rangle_{\mathcal{H}_+} + \langle f_-, g_- \rangle_{\mathcal{H}_-} \quad (2.18)$$

## 2.4.10 Spectral decomposition of Indefinite kernels

By spectral decomposition, we denote the matrix  $\mathbf{K}$  in terms of its  $\mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$  where  $\mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_N)$  is the diagonal matrix of eigenvalues  $\lambda$ 's and  $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_N]$  the orthogonal matrix of corresponding eigenvectors. We also assume  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda$ . The goal of the spectral transformation becomes one of applying a function  $f(\cdot)$  to the eigenvalues such that  $\lambda$  is non-negative and therefore the induced matrix  $\tilde{K} = U\tilde{\Lambda}U^T$  is PSD.

The following listed methods are some of the popular methods used in supervised learning settings to convert indefinite kernels to PSD.

### 1. Clip or Denoise

$$f(\lambda) = \max(0, \lambda)$$

The negative eigenvalues are regarded as noise [104] and are clipped to 0. This approach has sound theoretical backing where we consider the problem of approximating an indefinite kernel  $K$  with a PSD one  $\tilde{K}$  in Frobenius norm [256]. The problem is then formulated as follows:

$$\begin{aligned} \min e(\tilde{K}) &= \left\| K - \tilde{\mathbf{K}} \right\|_F^2 \\ \text{s.t. } \tilde{K} &\succeq 0. \end{aligned}$$

The solution to this formulation is calculated as  $\tilde{K} = U\tilde{\Lambda}U^T$ , where  $\tilde{\Lambda}$  is  $\text{diag}(\max(0, \lambda_1), \max(0, \lambda_2), \dots, \max(0, \lambda_N))$

## 2. Flip

We use the absolute values of the eigenvalues [104]. Very large negative eigenvalues become large positive values.

$$f(\lambda) = |\lambda|$$

Flipping the negative eigenvalues can be explained from the perspective of SVD decomposition of  $\mathbf{K}$ . As described in [256], let the SVD decomposition of  $\mathbf{K}$  be  $\mathbf{U}diag(\sigma_1, \dots, \sigma_N)\mathbf{V}^T$ , where  $\mathbf{U}^T\mathbf{U} = \mathbf{I}_N$ ,  $\mathbf{V}^T\mathbf{V} = \mathbf{I}_N$ , and the  $\sigma_i$ 's are permuted so that  $\mathbf{U}$  in this SVD decomposition is the same as  $\mathbf{U}$  from the spectral decomposition  $\mathbf{U}\Lambda\mathbf{U}^T$  of  $\mathbf{K}$ . Applying spectral decomposition, we have  $\mathbf{K}\mathbf{K}^T = \mathbf{U}\Lambda\mathbf{U}^T\mathbf{U}\Lambda\mathbf{U}^T = \mathbf{U}\Lambda^2\mathbf{U}^T$ , where  $\Lambda^2 = diag(\lambda_1^2, \lambda_2^2, \dots, \lambda_N^2)$  is the eigenvalue matrix of  $\mathbf{K}\mathbf{K}^T$ ; thus the singular value  $\alpha_i = \sqrt{\lambda_i^2} = |\lambda_i|$ . This means the induced  $\tilde{K} = \mathbf{U}diag(\sigma_1, \sigma_2, \dots, \sigma_N)\mathbf{U}^T$  becomes the transformed matrix.

## 3. Shift

$$f(\lambda) = \lambda + \eta$$

The kernel is made PSD by shifting the spectrum by the absolute value of the minimum negative eigenvalues. Compared to the flip and clip methods, the shift only affects the self-similarity and does not affect the off-diagonal elements. [256] showed the SVM dual formulation after shifting the spectrum of the kernel is equivalent to minimizing both the dual formulation before the shift and the 2-norm of the multiplier vector  $\alpha$ ; thus, the shift constant  $\eta$  can be regarded as the regularizer that penalizes the length of  $\alpha$ . A very large  $\eta$  decreases  $\alpha_i$  such that they are treated as non-support vectors (i.e. tends to zero).  $\eta$  plays a similar role to  $C$  in regularizing SVM.

## 4. Square

Rather than use the absolute values of the eigenvalues used with flipping, we square the eigenvalues. This is similar to squaring the kernel matrix.

$$f(\lambda) = \lambda^2$$

$\mathbf{K}\mathbf{K}^T = \mathbf{U}\Lambda\mathbf{U}^T\mathbf{U}\Lambda\mathbf{U}^T = \mathbf{U}\Lambda^2\mathbf{U}^T$ , where  $\Lambda^2 = diag(\lambda_1^2, \lambda_2^2, \dots, \lambda_N^2)$  is the eigenvalue matrix of  $\mathbf{K}\mathbf{K}^T$ ;

### 2.4.11 Additional kernel construction methods

The following are also valid kernels.

1.  $k(x_i, x_j) = k_1(x_i, x_j) + k_2(x_i, x_j)$
2.  $k(x_i, x_j) = \alpha k_1(x_i, x_j)$ , where  $\alpha \in \mathbb{R}$
3.  $k(x_i, x_j) = k_1(x_i, x_j)k_2(x_i, x_j)$
4.  $k(x_i, x_j) = f_1(x_i)f_2(x_j)$
5.  $k(x_i, x_j) = p(k_1(x_i, x_j))$ , where  $p(\cdot)$  is a polynomial function with positive coefficients.
6.  $k(x_i, x_j) = exp(k_1(x_i, x_j))$

Valid kernels will be created from base kernels with these methods

### 2.4.12 Normalizing a kernel

The data points in the feature space can be normalised to unit length via:

$$x \mapsto \phi(x) \mapsto \frac{\phi(x)}{\|\phi(x)\|} \quad (2.19)$$

Taking the dot product of the normalised points yields:

$$k(x, x') = \left\langle \frac{\phi(x)}{\|\phi(x)\|}, \frac{\phi(x')}{\|\phi(x')\|} \right\rangle = \frac{k(x, x')}{\sqrt{k(x, x)k(x', x')}} \quad (2.20)$$

## 2.5 Summary

In this chapter, we introduced the theoretical underpinnings of the proposed kernel framework. We stated the learning problem, defined linear functions and kernel functions. We explored the inner product and a Hilbert space with their properties. Mercer's theorem, Reproducing Kernel Hilbert Space, Cauchy-Schwarz inequality for kernels, metric spaces, and conditionally positive semidefiniteness are also stated. We explained the indefinite kernels with methods adopted to convert them into PSD. Lastly, how to generate additional kernels from basis kernels including normalising a kernel are also stated.

## Chapter 3

# Literature Review

### 3.1 Electronic Health Records (EHR)

The pervasive use of technology has emerged as fundamental to the delivery of quality healthcare services. To be precise, the deployed clinical information systems infrastructure in the UK plays a pivotal role by providing healthcare practitioners with the core platform to perform various tasks regarding routine patient care. To resist such infrastructure deployment in the healthcare domain would have been a strenuous objective in light of the rapid technological advancement at a low financial cost, its numerous benefits, and its subsequent successful ubiquitous use in several industries. The digitization process over the past few decades has electronically transformed clinical workflows and the passage of information within the health sector. As a consequence, there is currently a deluge of patient medical data, leading to “Big Medical Data.”

The initial adoption and promotion of technology for clinical use were primarily driven by the emphasis on modernising healthcare delivery [72], lowering and controlling the overall cost of care [114, 117], improving quality [24, 144], reducing errors, and ensuring patient safety [3, 24, 144]. In addition, the perceived possibilities of advancing biomedical and healthcare science through the reuse of the clinical data for follow-up and research that were seen at the onset of the digitization process also accelerated the transition from paper-based care [60, 115]. Literature suggests this move was necessary in order to overcome the problems associated with inefficient workflow management that were observed as obstacles to clinical decision making and assessment of outcomes [60]. A contrary view focused on the importance of its use in eradicating errors that emanate from issuing paper-based prescriptions as the significant contribution gained from adopting technology [3]. To elaborate further, errors such as the possibility of accidental overdose from handwritten prescriptions or erroneously prescribing a drug that dangerously interacts with existing medication were successfully eliminated with technology. As a consequence, prescription scripts are currently transmitted electronically to dispensing pharmacies, thereby speeding up the process and cutting out any further chance of errors.

The term “electronic health records” (EHR) is frequently used to describe electronic medical data from both inpatient (hospitals) and outpatient general practice (GP) IT systems. There has historically been little agreement on what type of IT system it refers to [3]. Nevertheless, the GP IT system used in a primary care setting in the UK is the primary focus of this study. “Primary health care” refers to care outside hospitals, and it is usually the first point of call for patients with non-emergency healthcare needs. It does not provide any form of specialised care. Some studies made a distinction between the definition of “electronic medical records” (EMR), which is considered an internal organisation system made up of patient clinical records collected and held locally, and “EHR” defined as an inter-organizational system that encompasses a broader set of information that

includes medical data from secondary care units [114, 129, 258]. Conversely, the systematic review of literature on mining electronic health records regards the definitions of EHR, EMR, and “electronic patient record” (EPR) as the same [126]. Likewise, this research will interpret EHR to include data captured during routine patient care at the primary care level and data from secondary care units such as specialist clinics, hospitals, third-party applications, and transfers from other GP IT system providers. EHR is, therefore, defined as the collection of “longitudinal” patient information spanning cradle-to-death medical data. The medical information combines demographic, lifestyle, and behavioural data with health records, thus providing a comprehensive view that coincides with the definition of patient-centred medical care [114]. The existing NHS IT infrastructure in the UK provides a fully integrated healthcare information system.

Embracing technology in healthcare delivery was initially a slow process, despite the anticipated benefits. According to a representative national survey conducted in the US in 2008, 17% of the 2758 physicians surveyed had EHR available, while only 4% had a fully functional IT system [79]. A fully functional IT system should have the following four basic capabilities: clinical documentation, result management, computerised order entry, and decision support [3]. The early adopters encountered a myriad of challenges ranging from the initial substantial cost, the perceived lack of financial return, the technical and logistical challenges involved in its maintenance, the consumer’s and physician’s concerns about the privacy and security of sensitive health information [37], physician apathy to technology [3], lack of education and training, and user resistance [98]. An analysis of the early stages of IT implementation concluded that technology did not lead to a significant improvement in resource utilisation, healthcare costs, or health outcomes [144]. It would have been difficult at the onset to measure such improvements. Although there were anecdotal reports and few formal cost-benefit analyses suggesting that EHRs provide financial benefits, Bar-Dayyan et al. provided actual evidence of a positive net financial return from using EHRs [24].

Recent developments addressed these areas of concern, leading to a reversal of apathy toward technology among key stakeholders. A few identified factors that influenced the acceptance of technology include the presence of in-house systems developers as users, integrated decision support and benchmark practices, the resolution of such contextual issues as provider knowledge and perception, the use of incentives, and legislation [144]. Categorically, government intervention played a significant role in driving up its use. In the USA, for instance, the Health Information Technology for Economic and Clinical Health (HITECH) Act, as part of the American Recovery and Reinvestment Act (ARRA) of 2009, earmarked \$19 billion US dollars to promote and encourage the adoption of EHR [37]. This figure had risen to \$ 27 billion by 2017 [114]. The HITECH Act applied the ‘carrot and stick’ approach to reward units that achieved “meaningful use” and penalise units that failed. For instance, extra pay was given for the “meaningful use” exchange of EHR with other healthcare systems as a means to facilitate and encourage interoperability within the care units. In terms of a penalty, units that did not use EHR meaningfully as specified incurred a loss of 1% of their Medicare fees in 2015, rising to 2% in 2016, and 3% in 2017 [37]. As a result of this approach, by 2012, nearly three-fourths of primary care physicians in the US were at least using EHRs for clinical encounters [48]. Prior to the HITECH Act, the US and Canada had implementation rates of 25% until 2009 [114]. In 2014, data from the American Hospital Association Annual Survey Hospital IT Supplement showed that the adoption rates of basic EHR in hospitals had soared to 59% [4]. In an earlier study, Adler-Milstein et al. showed that 75% of hospitals surveyed cited financial incentives as the facilitator for accepting technology [3].

In contrast, a slightly different approach was adopted in the UK. Competing primary care IT system providers were mandated by an agreement with the defunct General Practice System of Choice



(GPSoC) framework to show interoperability with third-party providers, among other key functional requirements. The agreement encouraged competition by allowing GPs in the UK to choose from the four GP system suppliers: TPP SystemOne, EMIS Web, InPS Vision, and Microtest Evolution. Failure to meet any of the agreement’s core requirements resulted in a financial penalty. The GPSoC framework was used by the NHS to fund the deployment of technology to local practices. It ended on the 31st of March, 2018 and was replaced by a more comprehensive framework called the GP IT Futures Systems and Services Framework. This new framework ensured the continuation, enhancement, and further expansion of the technological transformation of the IT infrastructure [41]. A typical GP IT system is required to have the capabilities displayed in Table 3.1 as specified by the framework [181];

Capability	Description
<b>Referral Management</b>	Allows for the inclusion of referral information in the patient record.
<b>Prescribing</b>	Supports the effective and safe prescribing of medical products and appliances.
<b>Recording Consultations</b>	Supports the standardized consultation recording and other General Practice activities
<b>Patient Information Maintenance</b>	Supports the registration and maintenance of all patient personal information.
<b>Resource Management</b>	Supports the management and reporting of Practice information, resources, staff members, and related organisations.
<b>Appointments management (GP)</b>	Supports the administration, scheduling, resourcing, and reporting of appointments.
<b>Appointments Management (Citizen)</b>	Allows citizens to manage their appointments online.
<b>Prescription Ordering (Citizen) appointments</b>	Allows citizens to request medication online and manage nominated preferred Pharmacies for patients.
<b>View Record (Citizen)</b>	Allows citizens to view their patient records online.
<b>Communication Management</b>	Supports the delivery and management of communications to citizens and Practice personnel.
<b>Digital Diagnostics</b>	Supports electronic requests and transfer of test results from other healthcare organisations.
<b>Document Management</b>	Supports the secure management and classification of all forms unstructured electronic documents.
<b>GP Extracts Verification</b>	Aggregated Data are extracted from clinical systems via the General Practice Extraction Service (GPES) and sent to the Calculating Quality Reporting Service (CQRS), both operated by NHS Digital.
<b>Scanning Medicines</b>	Support the conversion of paper documentation into digital format, preserving the document’s quality and structure.
<b>Medicines verification</b>	Allows compliance with EU Falsified Medicines Directive (FMD) for individually dispensed medications

Table 3.1: Capabilities accessible via the GP IT Futures Framework Capability

These also correspond to the minimum set of requirements specified by the “meaningful use” act for comprehensive EHR systems for hospitals in the US. The specification was defined according to the following categories; electronic clinical information (patient demographics, physician notes, nursing assessment, problem lists, medication lists, discharge summaries), computerized provider order entry (lab reports, radiology tests, medication, consultation requests, nursing orders), results management (view lab reports, view radiology reports and images, view diagnostic test results and images, view consultation reports), and decision support (clinical guidelines, clinical reminders, drug allergy results, drug-drug interactions, drug-lab interactions, drug dosing support) [52].

The NHS GP IT Operating Model [181] (first published in 2012) provides definitive guidance on digital services to be provided to GPs in the United Kingdom. The NHS recently published the General Practice Forward View in April 2016 with a commitment of an extra £2.4 billion a year to support GPs [180]. The package essentially aims to support struggling GPs by reducing workload, expanding the workforce, and investing in technology. The proof of the NHS’s commitment to applied healthcare technology infrastructure can be seen in the direct GP investment that will support better online tools and appointments, consultation and workload management systems, and better record sharing to support teamwork practises. The online services offer patients access to book appointments, order repeat medication, and view their medical records. These efforts demonstrate the organization’s dedication to the effective use of a fully integrated and functional technology infrastructure in care delivery. The result is evidenced by an increase in productivity, an improvement in patient care and experience, and the transformation of the working lives of NHS staff through the reduction of workload and improved morale [41]. Overall, the integration of fragmented information systems into the clinical life cycle will enable better patient management strategies and improve the quality and safety of care [16].

Interconnected systems enable a comprehensive patient EHR that is aggregated into a large data repository and accessible to care providers during routine care. The time-stamped data entities representing clinical and non-clinical events are stored in either a structured format using coded vocabularies, thereby retaining some consistency, or in an unstructured format that follows no specific organisational structure. The defunct Read Coded Clinical Terms dictionary in clinical terms version 3 (CTV3) or Read version 2 format that was used by the NHS since 1985 offered a comprehensive computerised thesaurus for recording structured data [182]. It was the standard dictionary of terminologies used by GPs in the UK to document patient findings until it was replaced with the Systematized Nomenclature for Medicine - Clinical Terms (SNOMED CT). Although other variants of clinical terms, such as the International Classification of Diseases (ICD), may also be used, the SNOMED CT is currently the most acceptable dictionary of terminologies used globally for coding EHR data [86]. The NHS dictionary of medicines and devices (dm+d) for prescribable medication and devices is also included in the SNOMED CT UK drug extension (UK version that is updated and published every four weeks). The recent international edition of SNOMED CT released on January 31, 2020, included 352,567 concepts [225]. These structured vocabularies enable clinicians to pick appropriate terminologies from a standard list when recording events. Structured data makes it easier and quicker to conduct searches or patient indexing. Its well-defined form also aids interpretability.

Despite their usefulness and ease of use, coded vocabularies can be restrictive and less expressive in terms of the information they convey. This is one of the downsides of the structured data form. The findings from Ye et al. [264] concedes that some traditional risk factors could not be directly captured by structured EHR data. With this limitation in mind, IT systems are currently designed with features that enable users to include information in unstructured data form. For example, digitised clinician notes are documented as free text narratives, while discharge certificates from hospitals and images are stored as attachments. The study [18] is a case in point, where the inadequacy of coded data was supplanted by using free-text clinical notes to document naloxone administration, thus the ability to classify the severity of opioid overdose as “moderate.” Free-text notes can also be used to overcome any bias that may be introduced when reimbursements are based on a stratified payment structure for using specific clinical codes, especially when few codes attract more payments [209]. Image capture is essential in healthcare since certain disease symptoms can only be effectively diagnosed with the aid of images. Common imaging data available in the EHR includes x-rays, computerised tomography (CT) scans, magnetic resonance imaging (MRI) scans, ultrasounds, echocardiography, positron emission tomography (PET), medical photography, and endoscopy, to name a few [129]. Codes from the

structured vocabularies are used to annotate the stored unstructured data, thereby making their inclusion searchable within the database. Other forms of unstructured medical data include videos, volumetric data, biomedical shape observations, whole-genome sequences, and pathology [83]

EHR contains highly interdependent biological, anatomical, and physiological time series. It consists of patient demographics, progress notes, problems, medications, vital signs, past medical history, immunizations, laboratory data, and radiology reports [114, 197]. EHR details a wealth of patient information that is essential to care management. Ultimately, the aggregation of these disparate data entities into a single repository provides the clinician with a one-stop holistic view of a patient’s health status, historical information, disease trajectories, and outcomes in order to reach a purposeful and informed clinical decision. It provides an abundance of medical information from a diverse patient population that can be used in population health management. However, given a functionally integrated care system capable of exchanging information across all care units, it is critical to note that the success of such systems is heavily reliant on effective data utilization [243]. In addition to the primary use of GP IT systems as a patient management system that is regularly accessed and updated during routine care, the stored EHR is often used for audit, quality improvement, especially chronic disease management, and health service planning [76]. Its secondary reuse outside of direct patient healthcare delivery provides an unprecedented opportunity and resource for comparative effectiveness research (CER), outcomes research, epidemiology, drug surveillance, and public health research [76, 116, 250]. The data offers a rich resource that contains complex interactions leading to hidden knowledge of intrinsic clinical value, such as correlations between diseases.

Unfortunately, the majority of EHR data are not analysed for valuable hidden knowledge [150]. EHR is primarily used to address the immediate needs of patients during routine care. It has been established that reliable and scalable EHR reuse has aided in drug discovery and biomedical research. [60]. It has led to phenotype discovery and is used to derive algorithms for longitudinal risk prediction [224]. Past literature suggests that such secondary use of health data can enhance healthcare experiences for individuals, expand knowledge about diseases and appropriate treatments, strengthen understanding about the effectiveness and efficiency of healthcare systems, support public health and security goals, and aid businesses in meeting the needs of their customers [39, 195]. Most importantly, EHR provides significant opportunities to build an evidence base on how to best manage multi-morbidity in various chronic diseases [75]. Although randomized controlled trials (RCTs) remain the gold standard for providing the strongest evidence in medical science research, the results from the trials can be replicated with EHR [116]. Additional validation of RCTs on real-life data strengthens the certainty of the clinical decision-making process; consequently, the focus has shifted towards mining EHR data for various uses.

## 3.2 Multivariate Analysis with EHR

Sophisticated multivariate analytical tools that can overcome the intricate characteristics of the data are required for mining EHR data. Researchers are currently applying “big data” analytics such as machine learning to extract knowledge and structure from the data. As decision-making situations become more complex, advanced techniques in healthcare delivery have gained popularity for addressing a wide range of problem types [87] particularly their use in extracting actionable insights that are highly beneficial for quality healthcare delivery. This is possible because we can transform raw EHR data into numeric features, thus converting the original problem into a mathematical problem that can be solved analytically or numerically. Healthcare analytics can help uncover insights that improve patient outcome prediction and treatment, clinical research, decision-making, prognosis, and patient management [99, 119]. For instance, the non-invasive use of technology in conjunction with

analytics can improve diagnosis, as evidenced by the case where analytics was applied to lung sound signals that were collected from low-cost microphones [87]. This provided an improvement and an alternative to over-reliance on the stethoscope for pulmonary sound auscultation in the diagnosis of asthma disease.

Furthermore, in the analysis of medication failure, Son et al. applied SVM to predict medication adherence in heart failure (HF) patients [227]. The study presents an illustration of the usefulness of multivariate analysis in an evidence-based approach to finding appropriate interventions for preventing medication failure. Understanding the risk factors associated with the lack of medication adherence in patients with heart disease is necessary to avoid worse outcomes. Wang, on the other hand, proposed a semi-supervised adaptive recursive tree partitioning (ART) framework for large scale patient indexing and search [243]. The model recursively built a tree structure based on solving an optimization problem with an objective function composed of both supervised and unsupervised terms. They presented a patient similarity evaluation approach for vectorized EHR data. Nevertheless, their method correctly retrieved patients with similar clinical or diagnostic patterns (from real-live data) that were likely to have congestive heart failure (CHF) within 6 months. Taken together, these results show that it is pertinent to leverage effective methods that can assuage the high cost of care, reduce the impact of worse outcomes, and improve the ineffective disease management seen with the traditional manual approach. This view is backed by the meta-analysis [238] of several machine learning methods used in predicting the presence of adverse events, estimating the subtype, and assessing the severity of HF. The seminal work showed that machine learning has played a pivotal role by significantly contributing to the management of HF.

As the burden of non-communicable disease (NCD) continues to exert undue pressure worldwide, the study [219] applied machine learning as an effective and efficient strategy to improve the participation rate for general health check-ups. Specifically, they used four predictive models based on the XGBoost, RF, SVM, and logistic regression (LR) algorithms to predict those unlikely to undergo general health check-ups. Despite the limitations of their approach, they demonstrated that machine learning predictive models outperformed existing heuristic methods for performing the same task. Existing research indicates that multivariate analysis of EHR can also help reduce the burden of long-term type 2 diabetes management. Areas of interest in the field of diabetic research include prediction and diagnosis, diabetic complications, genetic background and environment, and health care management. Consequently, several machine learning and data mining techniques are applied to perform these various tasks regarding diabetic care. For instance, [84] merged statistical modelling and medical domain knowledge with machine learning algorithms to assist personalised medical decision making using EHR from Columbia University's clinical data warehouse. The model estimated the optimal individualised treatment rules (ITRs) that were tailored according to subject-specific features. Accordingly, they constructed a decision tree for choosing the best second-line therapy for treating type 2 diabetes patients. The systematic review [131] showed that SVM was one of the most used machine learning algorithms in addressing problems regarding type 2 diabetes. Other recent examples include the ensemble-based approach of [8, 179], the Multi-Layer Perceptron (MLP), AdaBoost (AD), Trees Random Forest (TRF), Hidden Markov Model (HMM) of [199], k-Nearest- Neighbors, Naïve Bayes, Decision Tree, Logistic Regression [268], and the Gradient Tree Boosting (GTB) approach of [158].

The utility of multivariate analytics is also evident in its use to aid diagnoses in the mental health domain, specifically in the study of early detection of trauma survivors at risk of post-traumatic stress disorder (PTSD). Mental health disorders can have a significant impact on the quality of life of a suffering patient. Galatzer-Levy et al. applied machine learning to predict non-remitting PTSD

from information collected within 10 days of a traumatic event [96] and demonstrated the feasibility of machine learning-based feature selection and the predictions of non-remitting PTSD from early responses to traumatic events. Mental health diagnosis is still a complex endeavour. Nonetheless, machine learning methods may be especially suited to precision psychiatry's goals because they can directly translate complex pattern discoveries from the EHR into clinical relevance [44]. The central idea of the data-driven approach is to let the available data guide the investigation by extracting predictive patterns from the available heterogeneous data.

Likewise, by diverging from the traditional supervised methods of inferring precise phenotypes from EHR due to the limitations of pre-selected features and outcome labels, the study [143], motivated by deep learning, sought to apply an unsupervised approach to a phenotype discovery task. Their model produced continuous phenotypic features that accurately distinguished the uric-acid signatures of gout from acute leukaemia. The manual specification of entities that make up disease phenotypes is insufficient for extracting substrates that are relevant for the development of precision and personalised medicine. Another example of an unsupervised learning approach can be seen in the study [152], which applied a deep belief network (DBN) model learned by a contrastive divergence algorithm in order to identify the most informative risk factors that contribute to osteoporosis progression. By adopting a reconstructive learning strategy, they captured the original characteristics of the input data. The belief network used in the study applied two hidden layers to discover the underlying reasons behind the observed risk factors. The potential benefits of their work can be applied to early disease diagnosis and disease progression monitoring.

GP IT systems have evolved to become an essential part of daily routine care delivery; therefore, it is necessary to safeguard the accumulated sensitive EHR data. The UK's implementation of the General Data Protection Regulation (GDPR), covered by the 'data protection act' of 2018, specifies stronger controls for handling sensitive data. To ensure GP IT systems comply with this requirement, the NHS in England mandates that GP IT systems implement role-based access control (RBAC) for sensitive patient data. This measure is designed to ensure only users with clinical roles within the GP setting can access functionality that exposes patient-sensitive data while restricting non-clinical roles, such as admin or clerical roles, to functionalities that do not access the data. System security and privacy are integral parts of a fully functional system, and it is therefore pertinent to ensure these measures are not violated by regularly monitoring GP IT system access logs. This process can be automated by applying analytics to the logged data. Boxwala et al. illustrate this point with their seminal work that showed it was possible to apply machine learning (SVM and logistic regression) to detect spurious access to the EHR [40], thus showing it can be developed as a tool that can help security officers provide secured access to patient data.

The systematic review [213] ranked the commonly applied algorithms for spectroscopic cancer diagnosis. Their findings showed linear discriminant analysis (LDA) was used in 45%, SVM 10%, logistic regression 7%, principal component discriminant function (PC-DF) 7%, sparse multinomial logistic regression (SM-LR) 5%, and artificial neural networks (ANN) 5%. Other methods, such as classification and regression trees (CART), multinomial logistic regression (MNL), partial least squares discriminant analysis (PLS-DA), quadratic discriminant analysis (QDA), random forest, soft independent modelling of class analogies (SIMCA), and two-matrix discriminant analysis accounted for the remaining 20%. The review identified a few shortcomings, such as doubts over the validity of the studies that were based on small sample-sized data and the subsequent optimal performance rarely matched in clinical practise that was achieved with leave-one-out cross-validation (LOOCV). Bootstrapping was recommended as the approach to give more representative metrics. Similarly, [68] compared the performance of multiple machine learning models in predicting the likelihood of

hospitalisation. The study compared the performance of SVM, AdaBoost using trees as the weak learner, logistic regression, a naïve Bayes event classifier, and a variation of a likelihood ratio test on the medical histories of 45,579 patients that were extracted from Boston Medical Center (BMC). They successfully predicted 82% of the patients with heart diseases that were to be hospitalised in the following year. The outcome showed it was possible to identify and target patients at risk of hospitalisation with appropriate intervention, which can also help alleviate the increasing financial burden of hospital care.

Ye et al. used the XGBoost algorithm to develop a risk prediction model of incident hypertension within the following year [264] by using EHR data of over 1.5 million patients extracted from the Maine Health Information Exchange network. Their model (currently deployed for real-time live use) successfully stratified patients into five distinct risk categories, namely “very low,” “low,” “medium,” “high,” and “to very high.” Likewise, [233] applied logistic regression, naïve Bayes, and random forest algorithms to predict the risk and timing of deterioration in hypertension control. 1294 patients with hypertension were extracted from a chronic disease management programme at the Vanderbilt University Medical Center. The study showed that it is possible to predict the types of transitions in hypertension control using EHR data. Similarly, Arslan et al. compared the performance of various machine learning approaches in predicting ischemic stroke [13]. SVM outperformed stochastic gradient boosting (SGB) and penalised logistic regression (PLR) algorithms. Their study showed the effectiveness of applying analytical tools to predict ischemic stroke and extracting hidden relationships from EHR data.

Darabi et al. used gradient-boosted trees and deep neural networks in their study to predict the 30-day mortality risk of patients admitted to a single hospital’s Intensive Care Units (ICUs) [73]. They extracted 4,440 admissions from the MIT Lab for collaborative medical research (MIMIC III) data. The gradient-boosted trees outperformed the neural network due to the low number of examples used in their analysis. Noregeot et al. used a longitudinal deep learning model to predict disease activity in patients with rheumatoid arthritis (RA). They applied their method to data made up of 820 patients with RA extracted from two rheumatology clinics [183]. Their results showed that it was possible to use deep learning models to predict outcomes for patients with RA. There are two approaches to EHR analytics: static end-point prediction and temporal data mining [258]. Static end-point prediction seeks to model the relationship between the predictor variables and an outcome variable. Supervised learning in the form of classification and regression analysis are examples of static end-point prediction. Temporal data mining on the other hand, seeks to extract meaning from the longitudinal characteristics of the data. It is therefore imperative that we choose a representation and model that will enable us to include both spatial and temporal information in the analysis of heterogeneous longitudinal EHR data.

## 3.3 Problems with Modelling EHR Data

### 3.3.1 Heterogeneous EHR data

Modelling EHR presents difficulties for conventional analytical tools. The aggregated data is characterised as heterogeneous due to the disparate innate properties of its constituents. These diverse data types from different sources often contain complementary information that is useful in the clinical decision-making process. Nonetheless, finding a good set of risk factors for understanding disease characteristics and progression from the heterogeneous mixture is difficult [152]. Correctly identifying disease risk factors is critical in determining an appropriate intervention. Understanding medication treatment effects can also be a difficult task due to the nested hierarchies of temporal

heterogeneous relationships that are made up of overlapping single events, intervals, and sequences [145]. Multivariate, nested, and heterogeneous event patterns over time are difficult to make sense of. Regrettably, the uniformity promised by the use of structured data from coded dictionaries does not guarantee homogeneity. First and foremost, coded data designates a binary encoding indicating the presence or absence of an event. They may also encode categorical entities with or without an additional combination of real-valued and/or categorical attributes. Unfortunately, presenting EHR as categorical features causes the associated semantic meaning of clinical events related to a specific disease to be lost. Evidence of the heterogeneous nature of structured terminologies can be seen with the use of Read code “14L..00,” “H/O: drug allergy,” which indicates the presence of an allergy and has additional attributes specifying the particular drug the patient is allergic to and its severity. By the same token, Read code “246.00,” “O/E Blood pressure reading” is recorded with two additional real-valued attributes: systolic and diastolic blood pressure (BP), while Read code “136.00,” “Alcohol consumption” requires a categorical field with two levels (“Yes” and “No”) to indicate alcohol consumption and a numeric attribute for the number of units consumed per week. Numerical values can also be used to store nominal or ordinal variables, which therefore need to be differentiated from continuous variables since they would affect statistical analysis (for example, mean and variance) [120].

Despite these challenges, it is still necessary to utilise the full complement of clinical data in predictive modelling since the disparate items may yet hold significant discriminative value. For instance, the importance of modelling with a full complement of EHR data was recently demonstrated in a study [112] that applied machine learning to model mortality from unstructured free-text clinical notes. Firstly, the study showed that predicting mortality from clinical notes outperformed prediction using a fixed and predetermined set of physiological variables. Secondly, they demonstrated the predictive value of combining unstructured data with clinical time series, thus proving that clinical notes have an informative value, which is necessary for machine learning models to exploit. A survival analysis study [194] used a combination of Cox proportional regression and several machine learning models to predict the risk of survival in HF patients. Their findings revealed that models built on the EHR are more accurate, especially when comorbidities are included. Conversely, they also experienced difficulties dealing with both continuous and categorical variables in a single SVM model. Exarchos et al. presented a systematic and multi-parametric approach toward the prediction of oral cancer recurrence [89]. Using data made up of 41 patients with features extracted from imaging modalities (CT and MRI) of the head and neck region and gene expression data obtained from the cancerous tissue, the study applied Bayesian networks (BNs), artificial neural networks (ANNs), SVM, decision trees (DTs), and random forests (RFs), and achieved 100% accuracy. They addressed the problem of heterogeneous data by exploring the discriminative strength of each data source individually prior to combining the individual predictions to achieve a consensus decision, which led to the optimum performance seen.

In a study to identify patients at high risk for hyperlactatemia, Gultepe et al. applied both generative and discriminatory techniques to integrate heterogeneous patient data and form a predictive tool for the inference of lactate level and mortality risk [110]. They applied naïve Bayes, SVM, Gaussian mixture models (GMM), and the hidden Markov model to EHR data made up of 1492 patients from the University of California Davis Health System (UCDHS). The Gaussian mixture model was used to predict lactate level when the vital signs and white blood cell count (WBC) measurements were analysed in a 24-hr time bin, while SVM was applied to predict mortality in patients with sepsis. This was achieved with only three features: median lactate levels, mean arterial pressure, and median absolute deviation of the respiratory rate. Recently, the study [204] demonstrated the capability of recurrent neural network (RNN) in predictive modelling of the risk of

heart failure. The work addressed heterogeneity by using a neural attention mechanism to calculate the contribution score for each medical code within each patient’s visit. They applied the REverse Time AttentIoN model (RETAIN) RNN model with two GRU RNNS to generate attention weights for the purpose of interpretability. As a result of their findings, the shorter length and incomplete patient data posed a challenge in predicting the risk of heart disease. RNN models utilize a representation and transformation of the sequences into binary even length vectors equal to the size of the vocabulary representing the symbol or by padding the uneven-length sequences. In addition, evidence from the study suggests a large dataset may be required to yield superior classification performance.

Transforming raw heterogeneous entities to homogenise the variables could further exacerbate the problem with high-dimensional EHR data. The study [196] simultaneously addressed this problem and heterogeneity by applying recursive feature elimination (RFE), an embedded backward elimination strategy, to iteratively rank the variables via the clinical kernel that was derived from both categorical and numeric data. Several other works have applied the multi-kernel learning framework as a means of addressing the problem [153]. Although the study [153] used MKL to address heterogeneous pulse signals, several feature extraction methods (fiducial point-based spatial features (FP), auto-regressive model (AR), time warp edit distance (TWED), Hilbert–Huang transform (HHT), approximate entropy (ApEn), wavelet packet transform (WPT), and wavelet transform (WT)) were applied prior to developing appropriate kernels for the extracted features.

### 3.3.2 Irregularly sampled EHR data

Literature shows that dealing with EHR data that is characterised by a strongly irregular time sampling pattern poses a huge challenge for the modelling process [226]. This major concern is exacerbated by the way the data was initially recorded. Clinical encounters are not generally collected in a scheduled and controlled fashion. Rather, they are recorded out of sync, at the time when patients require care. Naturally, as patients do not fall sick at the same time nor share similar health records or trajectories, EHRs across the entire population will often be high-dimensional, sparse, noisy, inaccurate, and incomplete with lots of missing values. Incompleteness due to many missing values is usually the leading problem with the data, followed by inaccuracy and inconsistency [39]. Although the GP IT system used to record the data serves its primary purpose as a patient management tool that supports daily clinical workflow, it tends not to be efficient for the needs of researchers [72]. A complete and accurate dataset is highly important in clinical research, as missing data are hard to interpret [76] and makes EHR biased inherently [84]. An incomplete EHR will not contain relevant outcomes and covariates, such as lifestyle, family history, and environmental variables [224]. These are likely to be regarded as non-essential if unrelated to the specific reasons for a patient’s encounter. Consequently, tools such as the BlueBay CT Visionplus application that are implemented with Quality Outcomes Framework (QoF) templates become relevant in assisting GPs with prompts to record any missing data items that could earn the Practice extra QoF points.

Despite these efforts, issues with missing data continue to impede effective EHR reuse and can have a significant impact on predictive risk modelling performance [57]. The study [139], found a compelling gap in content between the data captured during routine care and the data required for patient eligibility assessment for clinical trials. Only 35% of the patients assessed for recruitment had complete EHR data. Similarly, missing values ranged from 4% to 46% in some of the variables used in the study [39]. Missingness must be addressed because incorrect interpretation may jeopardize the success of studies. Missingness in itself conveys information, and therefore, the following assumptions for missing observations may be adopted for mathematical convenience when describing probabilistic models:



- Missing at Random (MAR): The probability of a missing observation depends on the observed values but not on the missing values.
- Missing Completely at Random (MCAR): The probability of a missing observation does not depend on the observed nor on the unobserved measurements.
- Missing Not at Random (MNAR): The probability of an observation being missing only depends on the unobserved measurements.

It is necessary to fully understand the patterns of missing observations before choosing an appropriate solution. We can thus summarise EHR content as inaccurate, which therefore raises the question of the quality of the data. An emphasis on data quality is necessary if practitioners and researchers are to trust the outcome of their study. It may be difficult to establish what constitutes “good quality” data, especially in the context of its secondary use. Assessing data for its fitness-for-purpose would seem like a good strategy to adopt. Previous literature [251] identified the following five dimensions of data quality: completeness, correctness, concordance, plausibility, and currency, with an additional seven broad categories of data quality assessment methods: comparison with gold standards, data element agreement, data source agreement, distribution comparison, validity checks, log review, and element presence. Whatever dimension is considered, one of the primary barriers to effectively using EHR for research is data quality [60, 72].

First and foremost, EHR attributes in their native form require transformation into numeric features for onward processing with mathematical models. Selecting appropriate informative predictor variables pertaining to a specific disease, case of interest, or determining an outcome target can be a strenuous and time-consuming task. It is usually common for practitioners with expert knowledge to specify the clinically relevant predictor variables and inclusion criteria for selecting a cohort of relevant patients for a study [89]. Spurious and irrelevant variables can be ignored, which in effect reduces the dimension of the data. Consequently, this allows for a computationally cheaper and straightforward approach towards investigating a particular case; however, the effects of the unknown and unobserved complex inter-variable interactions are omitted. Likewise, it presents a means of sidestepping the sample irregularity and uneven size of the raw EHR since the outcome is a fixed length feature vector. Furthermore, human experts are prone to selection bias since two practitioners may end up with opposing views on what constitutes clinically relevant variables for a study. Non-expert based selection methods, on the other hand, will mitigate against such bias. Kim et al. addressed this particular problem by applying univariate analyses based on Kaplan-Meier analysis for categorical variables and univariate Cox regression for continuous variables in selecting variables for a breast cancer recurrence prediction model [136].

Evidence from studies has also shown that the expert-based feature selection methods have weaknesses due to their likelihood of discarding potentially useful information. A case in point is the study [232] that found predictive value from predictor variables excluded by expert knowledge-based methods. They showed there was no correlation between the excluded variables and expert-based selection, thereby proving indeed the discarded variables had informative value. Similarly, Lasko et al. demonstrated that expert-engineered features are no more discriminative than data-driven learned features [143]. Regardless of the approach adopted, the study features are derived from the heterogeneous mixture that may include categorical and real-valued measurements. A temporal abstraction method may also be employed to convert each time series feature value to a static representation, such as length, average, mean, slope, or the weighted sum of all values [19]. Likewise, similar aggregate features or representations as binary variables are commonly used to represent structured EHR data [56]. For example, diagnoses, procedures, and medications are represented

as binary variables indicating whether or not patients were assigned an ICD-9 code (International Classification of Diseases revision 9) or prescribed medication during a specified time window [223]. A quartile-based discretization method was then used to convert numerical measurements into binary features. This presented the traditional uniform-length tabular representation of the data without the temporal and spatial position of the variables of interest.

Transforming unstructured data is more challenging. Representation as a discrete sequence of symbols is one of the pliable solutions that can be adopted. A transformation into a featureless discrete sequence of symbols offers the opportunity to apply methods commonly used in natural language processing (NLP) and biomedical research, where it is common to categorise text or analyse a protein sequence, such as DNA. The bag-of-Words (BoW) method, also applicable to NLP tasks, that yields a histogram of data entities representing the frequency of occurrence for each patient may also be used. An alternative process referred to as vectorization encodes the symbols such that they are represented as points in a Euclidean space [94]. This approach, however, produces high-dimensional and sparse feature vectors. Conversely, a form of discretization may be employed to overcome a similar problem with a high-dimensional real-valued time series. This is evident in the case of Zhao et al., who explored time series discretization using the Symbolic Aggregate approxImation (SAX) with different  $\alpha$  values to obtain symbolic sequences [266]. A few other discretization methods that can be applied to achieve the same goal include the Piecewise Aggregate Approximation (PAA) [134], Discrete Fourier Transform (DFT) [5], and the Discrete Wavelets Transform (DWT) [50]. In contrast, Sun et al. represented sequences by first specifying an observation window for each featured concept, then aggregating all events of the same features within the observation window into a single value [233]. A greedy forward selection procedure was then used to determine which concepts to keep. The study [19] formalised a solution for detecting adverse drug events (ADEs) from complex EHR data by addressing the challenge of modelling with uneven length EHR by leveraging temporality and sparsity. The study applied a three-phase symbolic transformation of sparse and multivariate time series features into a single-valued feature representation and demonstrated the importance of temporality in effectively predicting ADEs from complex EHRs.

Furthermore, machine learning techniques can serve as alternative methods for extracting features from the data. Miotto et al. introduced an unsupervised patient representation that they called “deep patient” [169]. They created and implemented a three-layer stack of denoising auto encoders to capture hierarchical regularities and dependencies in data before predicting the likelihood of patients developing various diseases. Lasko et al. used Gaussian process regression (GPR) to transform the data into a continuous longitudinal probability density [143] prior to feature selection with a deep learning approach. The study addressed irregular sampling of uric acid values by using a time warp function, which uses a simpler transformation by shortening the longer intervals and lengthening the shorter intervals between measurements. The irregular frequency of the values occurs due to an active disease or treatment stage when it is necessary to measure more frequently as opposed to a regular or random schedule. This provided a simpler way of making the sequence closer to stationary. In another study [237], missingness due to irregularly sampled and unequal-length time series was equally addressed with Gaussian process regression (GPR). Glomerular filtration rate (eGFR) was re-sampled into even-length vectors prior to classification using KNN and SVM, thus addressing both problems with a single model while establishing if eGFR trends were stable or unstable. Likewise, Gong et al. [103] also used GPR to transform irregular and sparse EHR into continuous trajectories when investigating the temporal correlation between depression symptoms and suicidal ideation. Ye et al. applied a univariate correlation filtering step to remove features that were not directly related to the outcome variable [264]. They applied the Cochran-Mantel-Haenszel test to binary variables, the Cochran-Armitage trend test to ordinal variables, and univariate logistic regression to continuous

variables. They applied KNN for missing data imputation. Galatzer-Levy et al. applied a Markov boundary induction algorithm for generalized local learning (GLL) to select features with the highest predictive value [96]. Their method found features that were independently associated with the target.

To address the problem of uncertain and incomplete EHR, Marlin et al. adopted an unsupervised learning technique based on a probabilistic clustering model that was designed to mitigate the effects of temporal sparsity inherent in the data [160]. They applied a generative model that used a diagonal covariance Gaussian mixture model for real-valued data. Physiological time-series data made up of 13 features with known prognostic values was extracted from the data and used to predict mortality outcomes associated with patient episodes. The study highlighted the importance of exploiting temporal data to discover distinct, recognizable physiologic patterns with prognostic significance. However, one major drawback of unsupervised learning methods in phenotype discovery is their inability to incorporate current medical knowledge and directly handle missing or noisy data [245]. These issues are common, and thus some form of adaptation is required to successfully convert problematic EHR into meaningful and useful clinical concepts. Accordingly, Wanga et al. developed a novel knowledge-guided constrained non-negative tensor factorization and completion method for phenotyping, which they called Rubik [245]. They implemented a solution with built-in tensor completion that could significantly alleviate the impact of noisy and missing data. As a result, Rubik discovered more meaningful phenotypes. Traditionally, imputation methods have been the go-to method for dealing with missing values. Tian et al. proposed Multiple Imputation using Gray-system theory and Entropy-based Clustering (MIGEC) as a hybrid method for missing data completion [235].

In contrast, in the breast cancer study [267] on a complete set of EHR without missing values, the K-means algorithm was applied as a feature selection method in order to recognize the hidden patterns of benign and malignant tumours. The optimal number of clusters was established by finding the minimum validity ratio.

$$K^* = \arg \min_K \theta = \arg \min_K \frac{d_{avg}}{d_{min}}$$

$d_{avg}$  represents the average distance of each cluster to its centroid and  $d_{min}$  represents the minimum distance between any two centroids.  $\theta$  is the validity ratio for various cluster counts, which ensured the identified cluster was compact and isolated from others. A membership function was then applied to estimate the proximity of new data points to the symbolic tumours (centroids). This measure was treated as a new feature prior to classification with SVM. Their approach also reduced the dimensionality of the data. Despite the successful use of machine learning methods for feature selection, there are disadvantages to applying supervised and knowledge-based approaches to data representation in predictive modelling. They scale poorly, do not generalise well, and often miss opportunities to discover novel patterns and features; moreover, features learned by deep learning methods are not easily interpretable [169].

### 3.3.3 Others

Another critical issue with EHR is the inconsistency with which events are recorded using coded terminologies such as the SNOMED CT, which has many concepts, synonyms, and related sub-disease hierarchies [185]. Recognized solutions like practice guidelines that ameliorate any ambiguities and ensure uniformity of documented observations are not strictly adhered to [135]. Clinicians are likely to apply different codes when recording the same clinical event. An example of this can be seen with the prevalence of miscoding, misclassification, and misdiagnosis that has led to the erroneous documentation of Type 1 and Type 2 diabetes, leading to incorrect interventions in patients with diabetes [77]. Specifically, the systematic review found “miscoding” due to the use of vague disease codes that do not specify the type of diabetes or contradictory coding as the commonest errors in

57.1% (n=531) of the studies reviewed. Permitting the use of multiple diagnosis codes also makes phenotype discovery a challenging task [78]. These data inconsistencies are a common problem and exist because more than one model for expressing a specific meaning exists. For example, a clinician may choose to record “sigmoid colon” and another may choose “rectum” when recording the location of colitis disease [120]. Likewise, when recording hypertension, a clinician may use “eclampsia” or any of the 178 other diseases from SNOMED CT considered related to or subtypes of hypertension [86]. Practitioners from both primary and secondary care units are usually unaware of any consequences at the point of recording the data. This only becomes apparent later on when the patient condition worsens or during the secondary use of the data. On the other hand, a different perspective regarding the potential cause of inconsistency originates from the likelihood that any two clinicians may not follow the same line of inquiry when presented with similar symptoms during consultation. After all, clinical encounters usually involve an interactive question-and-answer session between the clinician and patient.

### 3.4 Elastic Distance Measures

In this section, we provide a literature review of the proposed edit distance based kernel framework. We start with a historical view of the development of the edit distance, the variants used, and examples of problem-centric modifications applied to address specific problems. We highlight the methods of computing the edit distance. In this section, we also introduce the kernel framework and edit kernels used to address specific problems. The literature review detailed in this section provides the background to our implementation of the methodology in this thesis.

Literature shows that statistical pattern recognition problems represented in vector spaces can be straightforwardly addressed with a plethora of mathematically sound and computationally efficient algorithms that make it easier to address the learning problem. Unfortunately, this simple representation is less expressive and unsuitable for dealing with a myriad of problems, specifically problems that can only be expressed in unstructured data form. For example, in speech recognition, image classification, document categorization, and biological and medical records. Dealing with problems where variables are represented as “sets” rather than real-valued vectors requires methods that can best extract the inherent structure with complex relationships from the data. By sets, we mean a discrete sequence of symbols or numeric values called strings and time-series data. Using feature extraction to convert symbols into numeric feature vectors can be costly and risk losing information, especially when dealing with sets of uneven length. Besides, the temporal order, structure, spatial position, and characteristics represented by the sequences are not fully captured by vector representation. Therefore, an alternative method capable of retaining as much information as possible regarding the problem is explored.

On the other hand, structural pattern recognition methods applicable to unstructured data allow us to utilise powerful and flexible representation formalisms [43]. For instance, establishing relationships between objects can be achieved with the use of a pairwise (dis)similarity proximity measure that can quantify their degree of closeness or unrelatedness. By dealing with symbolic data in its native form, we can preserve much detail about the problem and thus easily incorporate domain knowledge to treat identical or dissimilar objects in the same manner. For instance, we do not have to transform uneven-length objects; rather, we can utilise elastic proximity measures that can work directly on them. This flexibility offers an elegant way of retaining information. The classical proximity measures, such as Hamming or Euclidean distance used in statistical pattern recognition problems are unsuitable since they first require transformation into even-length vectors.

An elastic proximity measure can be defined as an optimization problem of two-dimensional warping that specifies a symbol-to-symbol correspondence between two subjected symbolic sequences [239]. If certain criteria are met, the pairwise proximity data matrix can be embedded in an Euclidean space. By this, we mean that there exists a set of vectors in a Euclidean space such that the mutual distances between the vectors are the same as the pairwise proximities. This process forms a bridge between featureless structural pattern representation problems and the tools available to represent problems in vector space. The following are a few examples of elastic distance measures: edit distance, dynamic time warp (DTW) and longest common subsequence. This work, on the other hand, will concentrate on edit distance.

### 3.4.1 Edit distance

Edit distance (k-difference problem), introduced by Levenshtein [149] is an elastic error-tolerant dissimilarity measure commonly used to quantify the distortion between two sequences of strings. It requires computing the minimum number of edit operations needed to convert one string into another. The idea is to define a cost that assigns a non-negative value to each edit operation. The process involves traversing both strings one character at a time and assigning a zero cost if both characters match. If, however, they differ, we recursively compute the cost of all edit operations and select the one with the minimum cost. Commonly used edit operations are "insert," "delete," and "substitution." The Levenshtein edit distance assigns a unit cost for all edit operations. This is known as the simple edit distance. If, however, a different cost for the edit operations is defined, then it is referred to as "generalized edit distance [175]." A small total cost indicates few edit operations are required to model the distortion between the strings, while a high total cost indicates strong distortions.

The origins of the edit distance can be traced to Levenshtein's investigations into the transmission and matching of binary signals in signal processing. Notwithstanding, its development and current use can also be attributed to research in three distinct communities, where inexact (approximate) string matching is required [175]. First, it is impossible to recover and match signals that may have been corrupted after transmission over noisy channels in signal processing. Secondly, it is impossible to retrieve an exact match of DNA sub-sequences after possible mutations in computational biology. Thirdly, typing or spelling errors in text categorization make it hard to find a string of text against a dictionary of words. These three domains have somewhat contributed to the popularity of edit distance as a viable proximity measure. A sequence is considered a match or similar if it is within k distance of another sequence, and dissimilar or not a match if the distance is greater than "k". Finding the best solutions to these problems contributed to the growth and development of edit distance.

Prior to the discovery by Levenshtein, a similar approach was explored in dictionary-based spelling correction tasks. Damerau identified that 80% of all misspelt words were due to an error from a single character [71]. The error that led to the wrong word was either due to a single omitted character, an inserted character, or a transposition with the next character. These can be said to be common typing errors that lead to misspelt words. Damerau's approach, combined with Levenshtein edit distance, therefore, allows for "insert", "delete", "substitution", and "transpositions" as additional edit operation. This merger of two communities is known as the Damerau-Levenshtein edit distance and has been used in numerous text correction problems. For example, the study [27] applied the Damerau-Levenshtein distance metric to create an error-correcting code over the space of words in the English language. Their study constructed a case-sensitive pass phrase system that could tolerate zero, one, or two spelling errors per word with no loss in security. The goal was for this to serve as a measurable, secure password for use in a password-based authentication or key exchange scheme. They showed the system could be made to accept pass phrases that were arbitrarily reordered and the security cost calculated.

The naive method of recursively computing edit distance is computationally expensive and scales exponentially. Due to its overlapping subproblems, computing the edit distance between two strings is best achieved with a dynamic programming optimization technique. This idea first appeared in computational biology, where the objective was to find the best alignment between a pair of sequences. The Needleman-Wunch algorithm [176] is one of the alignment algorithms applied to find similarities in the amino acid sequences of two proteins. It finds the optimum similarity between sequences by allowing gaps in the alignment. Although these gaps increased the number of comparisons, they avoided this extra computational bottleneck by excluding comparisons that did not contribute to the maximum score. In order to compute the alignment score, they applied a two-dimensional array to compare all possible alignments of pairs of amino acids represented by pathways. A unit value was assigned to every cell and pathway. The maximum match finds the largest number that would result from summing the cell values of every pathway. This introduced the dynamic programming technique to sequence analysis.

However, the idea to apply dynamic programming to compute the minimum edit distance, in particular, was proposed simultaneously by Wagner and Fischer [242] and Okuda et al. [186]. Dynamic programming combines solutions to simpler subproblems by breaking down the process into smaller, more manageable ones. It uses memoization to retain the outcome of each step before computing the next subproblem. The optimal outcome is achieved at each step, which therefore guarantees the best solution is found at completion. The computational complexity of an edit distance between two strings is proportional to the product of their lengths [242]. It has a time complexity of  $O(mn)$ , where  $m$  and  $n$  are the lengths of the two strings. One of its main advantages is its capacity to match variable-length sequences. Its use has enabled the modification of the edit distance algorithm. Thereby, it is suitable for solving various kinds of sequence matching problems.

Edit distance and its variants have matured as a useful tool in approaching several structural pattern recognition tasks. For example, in time series retrieval [54], computational molecular biology [30, 151], video event classification [23], shape recognition [70], learning the pronunciation of words in conversational speech [208], handwriting recognition [92, 165, 187, 215], Chinese relation extraction [53], large-vocabulary spoken-dialogue tasks [62], password correction [27], and monophonic music comparison and retrieval [146]. The variety of applicability includes multivariate analysis, such as classification, information retrieval, or clustering. It has proven to be an effective, elastic, error-tolerant tool. Other techniques, however, are similar to edit distance. For example, the longest common subsequence (LCS) can be viewed as an edit distance with only "insert" and "delete" edit operations at a unit cost. The Hamming distance, on the other hand, only allows "substitution" edit operations at a unit cost. The Hamming distance can only be applied to equal-length sequences.

Wei [248] tried to address the context-free nature of edit distance. In computing edit distance between strings, no systematic effort was made to exploit the coherence or statistical dependencies that exist within the local context [248]. Thus, they proposed a new edit distance based on Markov random field theory. They called this the Markov edit distance (MED). This edit distance variant took advantage of the patterns' local statistical dependencies to improve sequence matching performance. It is assumed that the use of MED offers an opportunity to also capture domain knowledge. Song et al. extended MED by applying Markov Random Field theory to Needle-Wunch distance [228], which they called the Markov Random Field-based Edit Distance (MRFED). They improved the performance of their method by combining the statistics-driven approach MRFED with a token-based distance function, Term Frequency-Inverse Document Frequency (TFIDF) into a hybrid distance function [228]. This hybrid distance function achieved better performance on datasets, whereas MRFED was inferior.

Applying bespoke edit operations that are tailored to the problem at hand increased the appeal of edit distance and encouraged its deployment into new areas. However, this usually comes at an additional computational cost. Therefore, new solutions need to find a balance between functionality and efficiency. Shapira et al. showed that simple edit distance with "move" operations is NP-complete [216]. They then presented a polynomial-time greedy algorithm for non-recursive moves, which achieved a  $O(\log n)$  approximation factor to optimality on a subclass of instances of a problem of size  $n$ . Cormode et al. achieved a sub-quadratic approximation of an edit distance with the "moves" edit operation [61]. They were able to achieve a near-linear time algorithm at  $O(\log n \log *n)$  approximation. Their work embedded strings into L1 vector space, making it possible to tackle a variety of problems where the distance measure is required. Farivar et al. proposed a new algorithm to compute the edit distance using a graphics processing unit (GPU) [90]. This solution is of particular interest, especially when dealing with large-scale data analysis problems. They modified the dynamic programming method in order to reduce the amount of storage and eliminate control flow divergence. By carefully managing memory usage and control-flow divergence, their algorithm performed better than an efficient multi-threaded, CPU-based implementation. Recently, Balhaf et al. applied GPU parallel implementations that utilise unified memory technology to speed up the computation of edit distance between two strings [22].

One other drawback of edit distance is the fact that the optimal path of edit distance is only determined by the number of errors [81]. The lengths of the sequences being compared are not taken into consideration. The same number of errors with shorter sequences will be irrelevant when matching two longer sequences. This anomaly may be more pronounced with certain types of problems. Therefore, a form of normalization is required. The amortized weight for a given edit sequence is the ratio of its weight to its length, and the minimum of this ratio over all edit sequences is the normalized edit distance [14]. Marzal et al. proposed a new algorithm to compute the normalized edit distance (NED) [165]. In general, the computation of properly defined normalized edit distances cannot be carried out using the standard algorithms that are used for computing edit distances. As a result, they came up with their own algorithm that introduced a linear increase in computational complexity when compared against the unnormalized computation of edit distance. Oommen and Zhang [193] improved the NED computation by creating a new algorithm. They also showed that NED can be computed by auxiliary measures introduced by the constrained edit distance [192].

Furthermore, Arslan et al. [14] reduced the worst-case time complexity of computing the normalized simple edit distance from  $O(mn^2)$  to  $O(mn \log n)$ . Though theoretical, they noted that fractional programming-based distance calculations would give good experimental results. Diez et al. used the Sum-of-Paths (SoP) formulation to normalise edit distance and the longest common subsequence [81]. As a result, all possible alignments between the two sequences were taken into account. Its computation favours low-cost alignments compared to the expected cost of all sequence alignments. A weight applied to these alignments ensured the optimal or near-optimal alignments were found. Their procedure provided a model-independent distance that avoided noise due to shorter irrelevant sequences. Experimental results showed the normalised SoP edit distance clearly outperformed the standard edit distance and longest common subsequence.

Bilenko et al. used the Needleman-Wunch algorithm in tackling record linkage problems [35]. Seni et al. tailored the Damerau-Levenshtein metric to more accurately compensate for the types of errors that occur in script recognition problems [215]. They introduced three additional edit operations, "merge", "split", and "pair substitute", in their algorithm. The need for these additional edit operations was motivated by the common errors in handwriting recognition tasks, where two characters can appear to merge to form another character. The standard edit operators cannot deal with this sort of error.

Lopresti et al. used "block" edit operations that enabled them to capture the high-level structure leading to the true alignment of two strings [156]. In the context of monophonic music comparison and retrieval, Lemstrom et al. presented a general framework for sequence comparison by dealing with variations of edit distance [146]. They defined a distance function that was based on local transformation, for which the costs were allowed to be context-sensitive. They also introduced the concept of a transposition invariant distance function. This was accomplished without the sequences being explicitly converted into interval encoding.

The q-grams technique [106] implemented with edit distances was successfully used to query relational databases. The technique relies on generating short substrings of length  $q$  called q-grams and processing them using string-based queries and searches. The q-grams can be used in conjunction with edit distance since when two strings are within a short distance, they usually have a large number of q-grams in common. The idea of using q-grams in approximate string matching developed from specifying conditions that could improve the efficiency of using dynamic programming. These conditions frequently dealt with q-length continuous substrings [234]. Anyway, Gravano et al. [106] implemented a solution that only considered matches within  $k$  distance of a query for searches on a relational database. This use of q-gram substrings also allowed them to extend simple edit distance to allow a cost for block moves that was independent of the length of the block. They proposed to allow the movement of a block of contiguous characters at a certain cost  $\beta$ . The study by Ann et al. [12] achieved better results in computing the cost of "block" edit operations. They considered character "insertion", "deletion", "block copies", and "block deletions" as restricted variations of edit operations. By applying these reasonable restrictions, they achieved the optimal solution within polynomial time.

Bozkaya et al. used a modified version of edit distance to deal with numeric data, considering two sequences of real values to be a match if the majority of the elements in the sequences matched [42]. They created a mapping among the non-matching elements to check if there were unacceptable deviations among them. Their method expected that matching sequences should have lengths that were comparable. Since they were dealing with numeric sequences, the insertion of elements into a sequence was done by interpolation, unlike the outright insertion used for symbolic sequences. Both sequences were considered matches if the distance was within a defined threshold. Alignment-based learning (ABL) [240] used edit distance to first find identical words in a group of sentences. The same words in sentences were found where no edit operation was applied. The algorithm works by comparing and aligning plain sentences by taking the entire sentence into account.

The use of a probabilistic approach to learn the optimum edit cost from the data introduced a new dimension to the edit distance application. Ristad et al. used a memoryless stochastic model to learn a string edit distance from a corpus of examples [208]. They applied this to the problem of learning the pronunciation of words in conversational speech and recorded a one-fourth error rate higher than the standard untrained edit distance. Their success made a case for the use of stochastic models in pattern recognition systems that applied edit distance. In the duplicate detection domain, Bilenko et al. presented a method for automatically learning string similarity from a small set of labelled examples [35]. They used expectation maximization to learn the edit costs from the training data for each field. Since the fields differed in terms of characteristics, they applied a weighting to the distance measure according to the contribution to the true similarity or dissimilarity of duplicate records. They also achieved superior performance compared to the standard fixed-cost distance metrics. Filali et al. used a dynamic Bayesian network (DBN) to learn the edit distance cost [92]. They followed the work by Ristad and Yianilos on applying a generative model to learn edit costs. DBNs belong to the larger family of Graphical Models (GMs) and are well suited for modelling stochastic temporal processes,



such as speech and neural signals [92]. They applied their method to a pronunciation classification task.

Mccallum et al. used an undirected graphical model for string edit distance learning in their work [166]. They used a conditional-probability parameter estimation method by learning from both matching and non-matching pairs. Unlike the generative models used by Ristad and Yianilos [208], they used a discriminative objective function to discover latent edit operations. They also introduced and used "delete-until-end-of-word", "delete-word-in-lexicon", and "delete-word-appearing-in-other-string" edit operations. These were suitable for the task at hand. Their approach, based on conditional random fields (CRFs), a finite-state conditional random field model for edit sequences between strings, showed positive experimental results on several datasets. It outperformed previous probabilistic approaches. Oncina et al. learned an unbiased stochastic edit distance in the form of finite-state transducers from a corpus of pairs of strings [187]. They learned a transducer independently based on the marginal probability distribution of the input strings. Their approach differs from the expectation maximization techniques commonly used in previous studies to learn a stochastic transducer. Their approach optimised the parameters of a conditional transducer rather than a joint one. As a result, they avoided inducing bias in the form of a statistical dependence on the input string distribution.

Bernard et al. applied a model to learn the structure and parameters of a so-called conditional edit transducer [33]. The transducers inherit the advantages of conditional models as described by [187]. This enabled them to overcome the statistical bias and the limitation on the expressive power of using a generative model to compute the edit cost. The idea to implement graph edit distance was proposed by Sanfeliu et al. [212]. In calculating the graph edit operation cost, Neuhaus and Bunke [178] approach the problem with a generative model. They derived the maximum likelihood estimation of edit operation distributions from a labelled set of graphs using the expectation-maximization algorithm. Experimental results showed their probabilistic model outperformed application-specific methods of achieving the same task. However, the computational complexity of graph edit distance remains a challenge. Riesen and Bunke [207] introduced an algorithm based on the Munkres algorithm for solving the assignment problem. This yielded a suboptimal solution at a substantially faster rate. Their new approach considered only local rather than global edge structure during the optimization process. It also treated the edit operations as independent from each other, which made graph edit distance feasible for graphs with up to 130 nodes. Empirical results from experiments showed sufficient accuracy.

The Euclidean distance, commonly used to measure the distance between vectors in equal-length time series, becomes insufficient when the lengths are uneven. They are very brittle [133]. As a result, Berndt and Clifford developed the dynamic time warping (DTW) technique [34] for a speech recognition task. It is a method that enables the comparison of one-to-many points. In other words, time series can be compressed or stretched. It uses a warping path, which maps or aligns the elements of both sequences such that the distance between them is minimized [34]. The mapping allows for repeating elements in such a way that both sequences are aligned. The goal is to minimize the potential warping path. The following constraints: monotonicity, continuity, warping window, slope constraint, and boundary conditions are used to constrain the warping window. This is necessary so we avoid pathological warping, where too many points are mapped to a single point. The Sakoe Chiba or Itakura parallelogram warping windows are common constraints that are usually applied [249].

DTW has been extended to several time-sequence domains, where elastic measures are required. It provides a means to incorporate temporal relationships. However, it is computationally expensive

and does not obey the triangle inequality, so it is not a metric. Researchers placed emphasis on methods of constraining the size of the warping window or on methods to speed up its computation. For example, Xi et al. [259] proposed the numerosity reduction technique to speed up one nearest neighbour DTW. Since it is not a metric, it is unsuitable for lower bounding techniques applied for faster retrieval of a search pattern; therefore, approximate solutions were proposed. Ratanamahatana and Keogh [205] debunked some of the myths surrounding the application of DTW to other problem domains. Results from empirical experiments supported their claim that DTW is well suited for a diverse range of domains. They, therefore, encouraged its use. The work by Ratanamahatana et al. introduced the Ratanamahatana-Keogh Band (R-K Band) [206], which allows for any arbitrary shape and size of the warping band. They achieved a reduced error rate and, at the same time, faster CPU time.

Unlike Bozkaya et al.'s [42] method, Chen et al. proposed an edit distance with real penalty ERP algorithm capable of supporting local time-shifting [54]. Lp-norms are incapable of dealing with local time-shifting in time series analysis. Nevertheless, their study combined the L1-norm with edit distance, called edit distance with real penalty (ERP). This can also be viewed as a variant of edit distance on real sequences (EDR) and dynamic time warp (DTW), except that it is a metric distance function. It attempts to combine the merits of both by using a constant reference point for computing the distance between gaps of two-time series [82]. Another study by Chen et al. introduced the edit distance on real sequence (EDR) [55]. Just like the ERP, this method is derived from the simple edit distance. It assigns a unit cost if the absolute value of the difference between elements of a pair of sequences is within a certain threshold. This approach offers a more robust technique because it can handle noise and outliers better by reducing the distance between a pair of elements to 0 and 1. Marteau created the time warp edit distance (TWED), which introduced and used a parameter to control the stiffness of the elastic measure along the time axis [161]. These have become valuable distance measures that can be integrated into the kernel framework.

### 3.4.2 Edit kernels

Applying edit distance with the K-Nearest Neighbour algorithm (k-NN) was the standard method commonly used in tackling pattern recognition problems. Although it originally enabled structural pattern recognition problems to be solved, edit distance is currently being applied with advanced learning algorithms. One of the methods that can work with the edit distance measure is the kernel method. Joachims' success in applying SVM to a text classification task [128] demonstrated the possibilities for integrating the kernel framework into sequence analysis. Researchers sought to extend the advantages of the superior classification performance obtained with discriminative models. While generative models were successfully applied to the analysis of uneven length sequences, Jaakkola and Haussler used the hidden Markov model (HMM) to develop the Fisher kernel [124]. The kernel method was extended to discrete structures such as strings, trees, and graphs by Haussler [113] and Watkins [246]. These seminal works on applying a discriminative approach to protein sequences and discrete structures opened the floodgate for the use of the kernel approach in dealing with all types of sequence analysis, thereby solving a greater range of problems.

Bahlmann et al. improved the HMM technique that was applied in solving the online handwriting recognition and classification task [20] by applying the dynamic time warp (DTW) in constructing a Gaussian-like kernel named the Gaussian DTW (GDTW) kernel. By adopting a discriminative approach, the study avoided the problem of sensitivity to modelling assumptions seen with generative models. This was a complete deviation from the integration of a generative model in developing kernels, as seen with the Fisher kernel [124]. Experimental outcomes showed comparable results to HMM-based models. Likewise, Shimodaira et al. also deviated from the method applied in the Fisher

kernel by embedding dynamic time alignment of sequences directly into the kernel function [220]. This approach resulted in an improved training time in comparison to the use of the Fisher kernel. Their kernel, called the Dynamic Time Alignment Kernel (DTAK), demonstrated comparable performance in hand-segmented phoneme recognition with HMMs. Leslie et al. introduced the sequence similarity kernel called the spectrum kernel [148]. They showed that string-based kernels incorporated into an SVM provided a viable and computationally efficient alternative method of protein classification and homology detection. This was a complete deviation from the generative and computationally expensive Fisher kernels previously applied to solve the same problem. Their method could classify test cases in linear time.

Lodhi et al., motivated by the new use of kernels, created the string subsequence kernel (SSK). This was another work that deviated from the feature vector extraction method implemented by the bag of words kernel. The use of a frequency distribution in the bag of words approach excluded information regarding the order of words. The subsequence kernel is derived from all subsequences of length  $k$  in the text that are not necessarily contiguous [154]. This approach allowed them to capture semantic information, and the experimental results showed improved performance in a text categorization task. Leslie et al. applied a string-based feature map and introduced a mismatch kernel in dealing with a protein classification task [147]. They outperformed the Fisher Kernel by calculating sequence similarity using shared occurrences of  $k$ -length subsequences and counting up to  $m$  mismatches. Saigo et al. proposed a new string kernel adapted to biological sequences, which they called the “local alignment kernel.” They measured the similarity between sequences by summing up scores obtained from local alignment with gaps in the sequences [211]. This approach incorporated the sequence alignment technique into a kernel framework.

Following the work of Saiga et al. in remote homology detection between protein sequences, [211] Cuturi et al. derived an alignment kernel by taking the smoothed version of the maximum score spanned by all possible alignments of two sequences [67]. This unfortunately has the problem of diagonal dominance, which does not generalize well to unseen examples. They applied the logarithm to the obtained kernel matrix in order to rescale the obtained values. This log alignment kernel improved performance. Further study by Cuturi [66] followed up on this work and later introduced alternative kernels that were positive definite, faster to compute, and more efficient in classification tasks. Aseervatham et al. proposed a semantic kernel for semi-structured biomedical documents [15]. It used the UMLS framework to incorporate semantic meanings during the similarity estimations between textual documents. This significantly outperformed the linear kernel and the multinomial naive Bayes classifier.

It was expected that valid kernel functions could be derived from these sequence alignment techniques, including the edit distance, meaning the kernels are symmetric and meet the Mercer condition of being positive and semi-definite. However, this was not the case. Cortes et al., in their work on rational kernels [62] showed that the edit distance in a kernel is not positive definite. The need for a positive definite requirement was necessary to achieve optimization convergence with the SVM algorithm. However, this may no longer be the case as it is possible to train an SVM classifier with an indefinite kernel [111, 155, 191]. Ong et al., proposed an SVM approach that dealt directly with indefinite kernels [191]. Loosli et al. presented a theoretical foundation for an SVM solver in Krein spaces [155]. Learning directly with indefinite kernels is referred to as learning in Krein spaces. There are other methods, such as spectrum modification [155], that can be used to transform an indefinite kernel into a valid one. Different forms of modification can be applied to the negative eigenvalues of an indefinite kernel. Another approach that has been proposed is to de-noise the indefinite kernel by treating the negative part of the kernel as noise [191].

Nonetheless, the construction of string edit kernels based on the edit distance is still an attractive one. The excellent results obtained by implementing sequence matching techniques using the kernel approach prompted the development of edit distance kernels, also known as edit kernels. The kernel functions can be derived directly by simply computing pairwise edit distance on all data points or indirectly by combining the pairwise edit distance with other mathematical manipulations in order to derive the kernel. The Gaussian-based model that replaces the distance measure with computed edit distance values is an example of an indirect approach. A parameterized version enables the construction of a valid kernel that is semi-positive definite. Li et al. used string edit kernels in computational molecular biology to predict translation initiation sites (TISs) in eukaryotic mRNAs [151]. They improved on the previous work by Zien et al., which used a polynomial kernel based on the weak hamming distance with the SVM algorithm to recognize TISs. Their study applied a parameterized edit distance to a Gaussian-like kernel function. They also gave a biological and probabilistic interpretation to their edit kernels.

In a shape recognition task, Daliri et al. first mapped the contours of shapes to be recognized into a string of symbols [69]. They compared the performance of the subsequence string kernel [154] against the performance of edit distance-based algorithms implemented with Procrustes analysis. The edit distance-based algorithm outperformed the SVM with the string kernel algorithm. Their study showed that successful recognition of shapes can be achieved through symbolic representation rather than through computationally challenging visual feature representations. However, this would suggest that setting the edit distance directly in an SVM kernel may yield promising results for the same task. Neuhaus et al. created string edit kernels [177] by first performing structural matching with edit distance instead of defining a kernel function directly on a pair of strings. The kernel function was defined with respect to the squared distances between strings and a zero string  $x_0$  selected from the training data. A pairwise similarity function defined on two strings  $x$  and  $x^*$  calculated the squared distance between patterns  $x$  to  $x_0$ , as well as between patterns  $x_0$  and  $x^*$ , in relation to the squared distance between  $x$  and  $x^*$ . They showed that every string can be represented by a unique vector  $x_0$  in the corresponding feature space. The dot product of these vectors in this space is equivalent to the kernel function. They also showed that the Euclidean distance in this feature space is equivalent to computing the edit distance of the respective strings. Their method can be applied to any distance-based learning algorithm for strings and graphs.

In solving the problem of relation extraction between named entities from Chinese texts, Che et al. defined a kernel over Chinese string representations [53]. They used the improved edit distance (IED) to compute the similarities between two Chinese strings. Their approach involved using a thesaurus to compute the similarity of Chinese words before computing the cost of an edit operation. The IED kernel was incorporated into a voted perceptron and an SVM classifier. They used this to extract person-affiliation from Chinese texts. Their kernel approach required fewer training examples than feature-based models in order to achieve the same performance. The use of a kernel function ensured the expensive feature selection stage used in relation extraction was avoided. Tian et al. applied an edit kernel with SVM in an intrusion detection system (IDS) [236]. They applied an edit kernel and the longest sequence-based kernel. The result from their experiment showed that although the sequence kernels outperformed traditional kernels like the RBF, the common subsequence-based kernel gave significantly better classification performance than the edit kernel. This was because, by admitting gaps in subsequence, the kernel gave greater weight to longer subsequences and counted the common subsequence of all lengths. The sequence information was used by both sequence kernels to detect anomalies.

Ballan et al. applied edit kernel learning in an event recognition task [23]. This approach allowed the integration of temporal information for video recognition with the bag of words approach. They used the Needleman - Wunsch edit distance in developing a Gaussian edit kernel. But unlike others, their version had no need for a parameter to make the kernel a valid one. Daliri et al. followed up their earlier work on string-based shape recognition [69], and applied the edit kernel directly to tackle the same problem [70]. They used the normalised edit distance version, which enabled their method to overcome the problem of partial occlusion. Bellet et al. adopted the idea that marginalised kernels allowed the computation of the joint similarity between two instances by summing their conditional probabilities to create edit kernels. They came up with an edit distance-based conditional distribution as a way of learning new string kernels. They showed that edit distance between strings could be computed by learning edit probabilities in the form of parameters of a stochastic state machine. Automatically doing so captured background knowledge and improved the performance of the kernel [30]. Zang et al. proposed and used the edit distance with a real penalty that was introduced by Chen et al. to develop a Gaussian ERP kernel [265]. They embedded this into a kernel difference-weighted KNN classifier and used it to classify pulse waveforms. Their method outperformed other pulse waveform classification methods.

Bellet et al. used an approach introduced by Balcan for establishing if an edit similarity is  $(\varepsilon, \gamma, \tau)$  good for solving a classification problem [31]. This method offered an alternative method for learning similarity measures that may or may not produce valid kernels. The problem was formulated as an L1-norm SVM that was solved with linear programming. Thereby, a sparse solution was obtained that outperformed the dense solutions from standard SVM in terms of classification accuracy in the presence of noise. Since the number of support vectors increases linearly with an increase in the number of examples, their approach will scale up to solving large data problems. Shin et al. introduced alignment kernels based on the generalized notion of alignments and the soft minimum approximation technique [221]. They used this to present an alternative means to derive kernels from edit distance. Their method provided an easy way to check condition for positive semidefiniteness. Although their alignment kernels outperformed some benchmark kernels, they displayed a poorer time efficiency. Jia et al. [127] built on the work of Zhang et al. [265] in the classification of pulse waveforms. They applied a Gaussian time warp edit distance kernel function with SVM in their study and achieved a lower average error rate than current pulse waveform classification methods.

Marteau et al. extended the work on defining kernels based on an aggregation of local alignment scores [162]. They constructed a recursive edit distance (or time warp) kernel (REDK) that is positive definite under certain conditions. Their approach required a recursive regularization term in order to prove positive definiteness. This led to simple, sufficient conditions that allowed the construction of a positive definite exponential REDK. They achieved this feat by replacing the min or max operator in the recursive equation defining the elastic distance by a summation  $\sum$  operator. The kernel sums up all the costs of the existing subsequence alignment path instead of keeping the best path and then adding weighting factors that can be optimised. This factor was used to favour good alignments while penalising bad alignments. Experimental results showed that the positive definite REDKs outperformed the indefinite elastic distances from which they are derived. They also showed proof that the edit distance, dynamic time-warping distance DTW, time-warp edit distance TWED, and edit distance with real penalty ERP were all indefinite. A Gaussian DTW kernel is not always a PSD kernel. It performed significantly worse for certain tasks [109], which may be an indication of the potential pitfalls of using indefinite kernels.

### 3.4.3 Indefinite kernels

Kernel functions that violate Mercer’s condition of positive definiteness are referred to as indefinite if the eigen decompositions of the resultant matrix  $\mathbf{K}$  have negative eigenvalues. Indefiniteness occurs because pairwise proximity measures, which mimic how humans categorise objects and allow us to incorporate prior knowledge, are not always positive definite, particularly when dealing with non-vectorized data. For example, structural data like strings and images. Indefinite kernels tend to model objects accurately and offer a good representation of the problems we seek to address. Empirical evidence [111, 263] suggests indefinite kernels outperform PSD kernels and therefore can be used to solve a diverse range of problems. Nevertheless, overcoming indefiniteness was regarded as a difficult task since kernel learning algorithms were not originally designed to work with non-PSD kernels. Besides, they pose a computational headache and are hard to interpret due to a lack of geometrical and theoretical understanding [111]. Positive semidefiniteness, on the other hand, induces a reproducing kernel Hilbert space (RKHS) and therefore implies the Representer Theorem holds. PSD also guarantees a convex optimization problem that converges to a unique solution. This is not the case with indefinite kernels.

Despite violations of the core requirements, we still need to address indefiniteness since some problems can only be represented as such. A few methods have been proposed and used by different scholars to address the issue. The problem of dealing with indefinite kernels falls into two categories. One line of thought considers modifications to the spectrum of the kernel matrix by manipulating the negative eigenvalues to make them non-negative, while the other considers methods that are insensitive to the violations and, therefore, learn directly from the indefinite kernel matrix without any modification. Kernel learning algorithms such as LibSVM are adapted to learn directly from indefinite kernels. Both methods have their strengths and weaknesses. Spectrum modifications imply we have a RKHS, which means we apply a wider range of traditional linear learning algorithms. However, it has some disadvantages since we could be manipulating or discarding potential information and therefore producing a different characterization of the original representation of the data. The second method, which can be further subdivided into learning a positive definite proxy kernel and an indefinite kernel extension [263] does not change the original kernel matrix. It does, however, require a sound theoretical explanation. One of the proxy kernel learning approaches simultaneously learned the support vector weights and a proxy positive semidefinite kernel matrix, while penalizing the distance between the proxy kernel and the original indefinite one [159], instead of directly minimizing or stabilizing a non-convex loss function. Invariably, the problem is formulated as a perturbation of the PSD kernel. They treated the indefinite kernel as a noisy observation of the true PSD kernel. Another case study [111] solved the non-convex problem directly by first embedding the data in a pseudo-Euclidean space and minimizing the distance between the convex hulls of the two classes of data.

Following the embedding in an infinite pseudo-Euclidean space known as a Krein space, an alternative form of learning occurs. The study [189] showed indefinite kernels induce a reproducing kernel Krein space (RKKS) rather than the RKHS, which therefore results in a stabilization problem instead of a minimization one. They proved a general Representer theorem for constrained stabilization and proved generalization bounds by computing the Rademacher averages of the kernel class. Krein spaces are indefinite inner product spaces that are endowed with a Hilbertian topology. This means a Krein space can be decomposed into two separable Hilbert spaces with their corresponding positive and negative inner products.

### 3.5 Multi-Kernel Learning

Kernels are representations of patterns in a feature space defined by a reproducing kernel Hilbert space. Each kernel function implies a different feature map, and hence a different representation of the structure we seek to exploit. The variety of techniques applied to define pairwise similarity functions between objects is one of the appeals of the kernel framework. We can adapt the human-level definition of similarity between objects to integrate every aspect of the problem domain into kernel functions, regardless of form or physical characteristics. Consequently, it is possible to inadvertently introduce incongruous representations, resulting in noisy feature spaces that yield poor classification performance; hence, it is a challenge to produce good-quality kernels beforehand. Besides, learning with a specific kernel can also be a source of bias [101]. This problem is exacerbated when dealing with situations in non-vector space, such as strings, where the derived kernels are the sole source of information regarding the patterns. It is imperative and sensible to utilize multiple kernel functions or parameterize kernels rather than rely on a single one [17].

The initial motivation underpinning the development and advancement of multi-kernel learning (MKL) methods stems from the desire to improve classification performance via model selection from multiple kernels. The kernels, in this case, were usually derived from the same set of features. Static methods, such as the kernel target alignment (KTA)[64] (see section 4.6.1) that do not require any training offer a principled way of determining good quality kernels from a given set of kernels. In contrast, tuning kernel parameters or training a sizeable number of kernels on a large number of data points using the traditional cross-validation method of model selection is presumably computationally infeasible. The training time renders the approach ineffective; therefore, the need for a proper approach to addressing this obstacle has resulted in the field of *kernel learning*. Can similar techniques, such as the wrapper and filter methods, adopted in feature extraction with supervised learning, be generalized to model selection within the kernel framework?

Fusing multiple base kernels to form a single super kernel is an essential form of MKL. This is achieved by finding the right combination of base kernels that maximises a generalized performance measure [262]. Multiple kernels to be combined are usually derived from the same set of features or from different modalities, thus also providing an effective means of tackling problems made up of heterogeneous data entities. The base kernels that constitute an MKL procedure can be derived from strings, real-valued entities, and categorical entities. Linear separability is feasible in the new feature space created by concatenating multiple feature spaces. As highlighted in section 2.4.11, the linear sum and non-linear product kernels that combine multiple kernels are still valid kernels. Unfortunately, the new single super kernel may not fully consider the contributions of the base kernels in achieving the classification goal, especially if the kernels contain complementary information or are derived from orthogonal features. Therefore, a principled way of achieving the goal is needed, such as the weighted linear (conic) or convex sum of the base kernels.

The formulation of MKL is given as;

$$K_d(x, y) = \sum_{m=1}^M d_m K_m(x, y) \quad (3.1)$$

Where  $d_m \in \mathbb{R}$  for a linear combination of base kernels. However, in practice, further restrictions such as  $d_m \geq 0$  and  $d_m \in \mathbb{R}$  for a conic combination, while for a convex combination  $d_m \geq 0$  and  $d_m \in \mathbb{R}$ ,  $\sum_{m=1}^M d_m = 1$  are placed on  $d_m$  to yield interpretable solutions.

The learning objective of the MKL formulation becomes one of finding the optimal weighting coefficients  $d_m$ . Several techniques have been proposed in this regard. One of the key approaches incorporates this objective as part of the SVM dual formulation and then jointly learns both the support vector Lagrangian and weighting coefficients at the same time as a one-step strategy [17, 142]. However, the Second Order Cone Programming (SOCP) in [17] and Semi Definite Programming (SDP) in [142] are not computationally efficient and thus cannot process a very large number of kernels [262]. The one-step approach has the disadvantage that the joint SVM-MKL formulation is harder to solve or interpret, because it is neither differentiable in the dual form nor yields smooth functions. Alternatively, a two-step strategy, such as the one adopted in [202], can alternate between solving the SVM problem and then solving the weighting coefficient problem until convergence. It decouples the MKL weight coefficients from the SVM objectives, leading to the use of off-the-shelf SVM solvers to find the solution to SVM before solving for the coefficient in an alternating manner until convergence. Rakotomamonjy et al. applied the subgradient-based method to solve the problem of weight coefficients [202]. Although the two-step strategy is intuitive, it may not be computationally efficient to alternate between SVM optimization for each iteration and then updating the coefficients until convergence.

The average sum kernel, on the other hand, offers a variant of the sum kernel that is unconcerned with the weight coefficients. It simply computes the average of the kernels to be combined.

$$K = \frac{1}{2}(K_1 + K_2)$$

If both kernels are similar, then the resultant sum kernel is equivalent to either kernel. The study [74] applied the average kernel as the basis for formulating an MKL method that incorporated the difference between the two base kernels. The average kernel is reformulated into

$$K = \frac{1}{2}(K_1 + K_2) + f(K_1 - K_2)$$

The function  $f(K_1 - K_2)$  provides a meaningful way of integrating the difference between  $K_1$  and  $K_2$ . They provided three functions  $f$ : absolute value (AV), squared quantity (SQ), and squared matrix (SM). Their formulation used the parameters  $\tau > 0$  to control the effect of  $f(K_1 - K_2)$  and  $Y = \text{diag}(y_i)$  as the diagonal matrix made up of  $y_i$  labels. However, cross-validation is still required to choose  $\tau$  and the kernels derived with the AV and SQ methods are not guaranteed to be PSD. Drawing inference when presented with an unlabelled example  $x$  requires the use of a target label. The method computes the test kernel  $K(x, x_i)$  twice by first setting  $y$  to -1 and then to 1. The new example is assigned to the class whose distance from the SVM hyperplane to  $x$  is greater. In addition, the process is computed recursively for multiple kernels.

It is evident that the application of heuristics, as highlighted in addressing MKL, falls short of the robust solution required to address more challenging problems. A more efficient method, such as automating the process, was achieved in part by defining a reproducing kernel Hilbert space on the space of kernels known as hyperkernels, which resulted in a statistical estimation problem similar to minimizing a regularized risk functional [190]. Following the previous work [188], the convex optimization problem was expressed and solved as a semidefinite programming problem (SDP). Semidefinite programming optimizes a convex function over the convex cone of positive semidefinite matrices, or convex subsets. The concept established an empirical quality functional  $Q_{emp}(k, X, Y)$ , where  $k$  represents the kernel functions  $K(x_i, x_j)$ ,  $x_i, x_j \in X$ . KTA, regularised risk, and the negative log-posterior can all be used to define the empirical quality function  $Q_{emp}$ . It is an indication of how well  $k$  is specific to dataset  $X, Y$ . It is possible to find a kernel with minimum  $Q_{emp}$  given a diverse class of kernels defined on  $X$  with labels  $Y$ . Similarly, Lanckriet et al. [142] also learned a kernel matrix



from data through the semidefinite programming technique. In a transductive learning manner, they learned a kernel matrix corresponding to the entire dataset by optimizing a cost function that depended on available labels. Restricting the weight coefficients to ensure a conic combination of base kernels yielded a quadratically constrained quadratic programme (QCQP) problem that could be solved with SDP. Despite the earlier gains made using this method, its computational infeasibility on big data problems was a major drawback. Nonetheless, the methodology proposed by Lanckriet et al. sparked other works aimed at improving its shortcomings.

For instance, Bach et al. addressed the problem of the non-differentiable cost function in the formulation by Lanckriet that rendered SVM techniques that are robust to large data, like sequential minimal optimization (SMO), untenable [17]. Although the MKL formulation is convex, it leads to a non-smooth function and therefore poses a challenge since convergence is not guaranteed. They added a Moreau-Yosida regularization term to the primal objective function and introduced a support kernel machine (SKM) that was motivated by solving a block-based variant of SVM. The input is first decomposed into  $m$  block components, such that  $x_i = (x_{1i}, \dots, x_{mi})$  and the SVM weight vector  $w = (w_1, \dots, w_m)$ . To solve the soft margin problem, they minimized the square of a weighted block  $l_1$ -norm, where, within every block, an  $l_2$ -norm is used. Unlike  $l_2$ -norm constrained variables,  $l_1$ -norm constrained variables produce sparse solutions. Since this was not sufficient for the SMO algorithm, they added the regularization term to yield an SKM primal formulation. Sonnenburg et al. [229] recast the problem as a semi-infinite linear programme (SILP), which they generalized to a larger class of problems, including regression and one-class classification. Following the idea of  $l_1$  constraint [17], their solution promoted a sparse combination of the base kernels, thereby offering interpretability of the decision function that was lacking with earlier non-sparse solutions.

Xu et al. formulated a closed-form solution for optimizing the kernel weights based on the equivalence between group-lasso and MKL [262]. Their motivation was driven by the need to find an alternative means devoid of any dependence on commercial applications or complicated techniques devised by earlier studies to achieve the same goal. By generalising to the case of Lp-MKL ( $p \geq 1$ ), they provided a unified solution for the family of Lp-MKL models. L1-norm solutions may not yield optimal classification performance when compared to unweighted combinations of the kernels due to the potential for discarding kernels that may contain complementary information. Conversely, Kloft et al. [137] on the other hand, observed the  $l_1$  - norms that yield sparse solutions rarely outperformed baselines in practical cases, despite the time saved by not evaluating the entire kernel functions at the testing stage. Sparse solutions do not generalize well. As a result, they extended MKL to arbitrary norms, i.e.,  $l_p$  norms with  $p \geq 1$ , by devising two efficient interleaved optimization strategies, which rendered costly semi-infinite and first, or second-order gradient methods obsolete. In a non-linear approach to learning kernels, [63] extended the MKL problem to learning a polynomial combination of base kernels for regression by simplifying the optimization problem from a minimax problem to a simpler minimization problem and proving the solution always lies on the boundary.

### 3.6 Summary

This chapter has attempted to provide a brief summary of the literature relating to EHR setup in the context of primary care settings. It describes the issues experienced with the adoption of technology in healthcare delivery and the subsequent use of incentives to drive up its use. The makeup and contents of the primary tool used by care providers in the management and delivery of care are also described, leading to the accumulation of longitudinal big medical data with potential intrinsic value. The literature revealed the numerous benefits and problems associated with the secondary use of the data, especially since the data was not originally collected for research purposes. The use of supervised

machine learning in addressing healthcare problems is also illustrated with a few examples. The issues of data representation and feature extraction are highlighted, along with some of the methods adopted in addressing them. A brief history of the development of edit distance in three different disciplines is stated. Examples of its variants and method of computation are stated. Its use in the context of the kernel framework is also described. The literature identifies several multi-kernel learning techniques and methods adopted in addressing indefinite kernels. In conclusion, this review demonstrated that multivariate analysis of problematic EHR has numerous advantages.

# Chapter 4

## Methodology

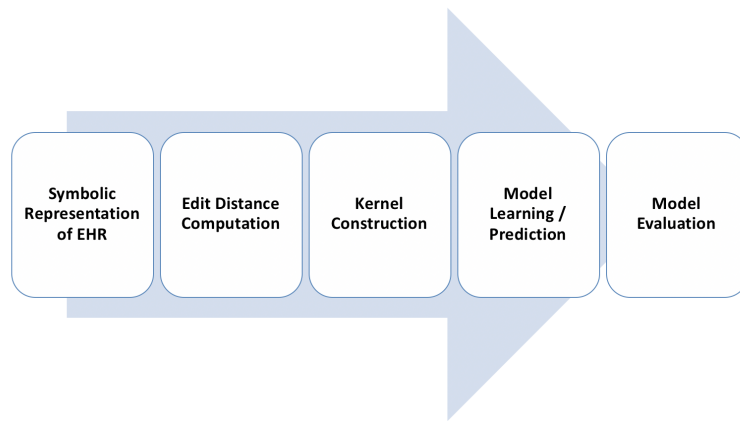


Figure 4.1: The kernel learning pipeline

### 4.1 Overall Study Design

#### 4.1.1 Independent variables

Data entities describing a multitude of clinical and non-clinical events stored within the 6 data tables for which entries are recorded for each patient were assessed for representation in the model as predictor variables. The data was extracted from the Clinical, Repeat, Therapy, Consultation, Referral, and Test relational data tables. The contents include diagnosis, procedures, conditions, referrals, recalls, therapy items, and real-valued test results. Most of the data were recorded with read codes, except for the therapy items that originated from a drug dictionary. Nevertheless, each entity has unique attributes and characteristics that make the entire mixture heterogeneous.

#### 4.1.2 Dependent variables

The experimental study's goal was to predict who is at risk of developing type 2 diabetes based on a history of high blood pressure. As a result, those who had a prior episode of elevated systolic BP of 130 mmHg and diastolic BP of 80 mmHg recorded prior to being diagnosed with type 2 diabetes are represented as the positive outcome variable and assigned the (+1) label. Patients with readings less than 130/80 mmHg before they developed type 2 diabetes were regarded as having a "negative" outcome and were assigned (-1) label.

## 4.2 Multivariate Analysis

In this section, we introduce the edit distance dissimilarity measure and variants of edit kernels developed from it. The kernels were derived from a symbolic representation of the data. We took advantage of the capacity of bespoke variants of the edit-distance kernels to treat arbitrary-length temporal sequences of symbolic data on the same computational level as real-valued data in a manner not possible within standard (non-kernelized) machine learning approaches. This study explored ideas in an empirical search for the optimal classification performance of the derived kernel matrices. Multiple kernels were combined algebraically with the SimpleMKL [202] algorithm (see section 4.5).

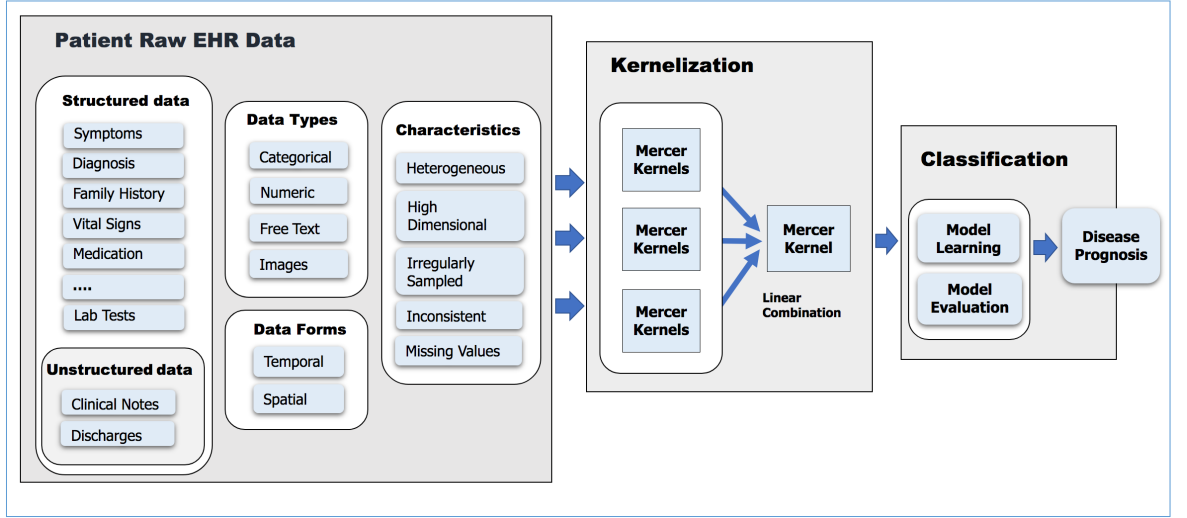


Figure 4.2: The proposed kernel framework for disease prognosis modelling with EHR data

### 4.2.1 Edit distance and variants

Edit distance is a more general and accurate measure of sequence dissimilarities [151]. It requires computing the minimum number of edit operations needed to convert one sequence into another. Commonly used edit operations are insert, delete, and substitution. A non-negative value is assigned to each edit operation, and the minimum total cost of transforming one sequence into another is selected. The process involves traversing both sequences one symbol at a time and assigning a zero cost if both symbols match. If, however, they differ, we recursively compute the cost of all edit operations and select whichever has the lowest cost.

Edit distance in its native form may not be suitable to address all types of problems; therefore, we created the additional variants with some small modifications. These are computed using dynamic programming in the same way.

#### 1. Edit distance

Given discrete sequences,  $x = x_1, \dots, x_i$  and  $y = x'_1, \dots, x'_j$  derived from Alphabet  $\Sigma$ , the edit distance between the two sequences  $d_{ed1}(x, x')$  is computed recursively via dynamic programming applied to the following equation [242].

$$d_{ij} = \min \begin{cases} d_{i-1,j} + 1 \\ d_{i,j-1} + 1 \\ d_{i-1,j-1} + (if\ x_i = x'_j\ then\ 0\ else\ 2) \end{cases} \quad (4.1)$$

where  $d(.,.)$  is the distance between a pair of sequences and  $x_i, x'_j \in X$ .  $d_{ed1}(.,.)$  denotes the native edit distance. It is also a metric as it obeys the four axioms of a metric space.

## 2. Edit distance with length normalisation

Normalization has been proven to enhance the possibility of finding patterns in data. By applying a normalising factor  $N$  to the  $d_{ed1}$  distance computation, a variant of the edit distance is created. Normalizing with the length of the longer sequence takes into consideration any effect the length of the sequences may have on the proximity of a pair of data points.

$$d_{ed2}(x_i, x'_j) = \frac{d_{ed1}(x_i, x'_j)}{N} \quad (4.2)$$

where  $N$  is the length of the longer sequence.

## 3. Edit distance normalised by number of common items

We normalise the native edit distance  $d_{ed1}$  computation by the number  $|x_i \cap x'_j|$  of common elements between both sequences.

$$d_{ed3}(x_i, x'_j) = \frac{d_{ed1}(x_i, x'_j)}{|x_i \cap x'_j|} \quad (4.3)$$

## 4. Edit distance normalised by exponent of number of common items

The normalisation factor applied to the native edit distance  $d_{ed1}$  is scaled exponentially to  $\lambda = 2^{|x_i \cap x'_j|}$

$$d_{ed4}(x_i, x'_j) = \frac{d_{ed1}(x_i, x'_j)}{\lambda} \quad (4.4)$$

**Justification:** The motivation for these variants of edit distance,  $d_{ed2}$ ,  $d_{ed3}$ , and  $d_{ed4}$ , was to apply a form of normalization to the computed distance measure. An attempt was made to factor in the effect that the sequence length and number of identical symbols may have on determining how objects are similar or differ. The length of the longer sequence, the number of common items in both sequences, and an exponential form of the number of common items are used as the normalising factor.

## 5. Subsequence edit distance

Finding relationships by exploiting the (dis)similarity between subsequences of symbolic EHR data allows us to model proximity in terms of items recorded during a single consultation. The subsequences for this class of distance measures are derived based on grouping items recorded in a single consultation, i.e., items with the same event date. The distance between a pair of sequences is computed by summing the edit distances between the subsequences contained within the sequences.

Let  $X$  be a sequence with  $m$  subsequences and  $X'$  be a sequence with  $n$  subsequences, with  $X = [(x_1, t_1), \dots, (x_m, t_m)]$  and  $X' = [(x'_1, t_1), \dots, (x'_n, t_n)]$ . The distance between these two sequences  $X$  and  $X'$  can be calculated as follows:

$$d_{ed5}(X, X') = \sum_1^{mn} d_{ed1}mn(x_m, x'_n) \quad (4.5)$$

**Justification:** We assume that two patients are nearly identical when their records show similar collective events were recorded per consultation.

## 6. Subsequence edit distance normalised by edit distance between the pair of sequences

This formulation is similar to the subsequence edit distance described above. In this case, however, we normalise the computation using the edit distance between the full length pair of ungrouped sequences.

$$d_{ed6}(X, X') = \frac{d_{ed5}(X, X')}{d_{ed1}(X, X')} \quad (4.6)$$

where  $d_{ed1}(X, X')$  is the normal edit distance between sequence  $X$  and  $X'$

#### 7. Subsequence edit distance normalised by the number of common symbols

$d_{ed7}$  normalises the computed subsequence edit distance  $d_{ed5}$  by the number of items common to both sequences.

$$d_{ed7}(X, X') = \frac{d_{ed5}(X, X')}{N} \quad (4.7)$$

Where  $N$  is the total number of common items in both records

#### 8. Subsequence edit distance normalised by the number of items common to each pair of subsequences

$d_{ed8}$  Normalizes the distance between subsequences with the number of items common to both subsequence

$$d_{ed8}(X, X') = \sum_1^{mn} \frac{d_{ed1}mn(x_m, x'_n)}{N(x_m, x'_n)} \quad (4.8)$$

where  $N(x_m, x'_n)$  computes the number of items common to the subsequences.

**Justification:** If indeed we can establish similarity from the collection of subsequences grouped by items recorded per consultation, we considered applying a normalizing factor that sought to incorporate the similarity in terms of how identical the entire sequences are without any form of grouping. The distances  $d_{ed6}$ ,  $d_{ed7}$ , and  $d_{ed8}$  apply various forms of normalization to the subsequence edit distance  $d_{ed5}$ .

#### 9. Edit distance computed with the number of unmatched and matched symbols

The number of similar items in a pair of sequences should play an important role in determining how similar the items are. Likewise, the number of unmatched items. This modified edit distance  $d_{ed9}$  is combined with a decay factor that accounts for any effects these may have on the distance calculation. The proximity between a pair of sequences decreases as the number of matching items increases, while the distance increases by a factor of the number of unmatched items.

$$d_{ed9}(x_i, x'_j) = \lambda * d(x_i, x'_j) \quad (4.9)$$

where  $\lambda = \frac{\tau^u}{2^m}$ ,  $u$  = number of unmatched items and  $m$  = number of matched items. The value for  $\tau$  is set by cross validation.

**Justification:** The idea for this normalising step is motivated by the approach used in [70] to transform a pair of shape contours into a string of symbols. In this research, we use this to factor in the ratio of matched items to unmatched items. Two sequences should be a lot closer if they have more items in common than apart. Finding the value for  $\lambda$  is achieved via cross validation.

### 10. Edit distance with controlled equality

This edit distance variant  $d_{ed10}$  is realised by applying the controlled equality concept of the edit distance to the real sequence [55]. In this case, we relax the equality between a pair of symbols (Therapy) by including the number of elapsed days between medications as a threshold for determining equality of symbols. After a specified number of days  $h$ , a pair of similar therapy items are no longer treated as equal. The following formula is derived from a pair of sequences  $x = x_1, \dots, x_i$  and  $x' = x'_1, \dots, x'_j$ , with dates  $T = t_1, \dots, t_i$  and  $T = t_1, \dots, t_i$  and  $S = s_j, \dots, s_j$ :

$$d_{ed10}(x_i, x'_j) = \min \begin{cases} d_{i-1, j} + 1 \\ d_{i, j-1} + 1 \\ d_{i-1, j-1} + (if\ x_i = x'_j\ and\ |t_i - s_j| < h\ then\ 0\ else\ 2) \end{cases} \quad (4.10)$$

The native edit distance is modified to add this constraint. **Justification:** This distance is motivated by the need to use the number of days between medications as a threshold to control the similarity of prescribed medications. Similar medications will not be treated as equals with a zero edit distance cost if the gap between them exceeds this threshold.

### 11. Edit distance on real sequence (EDR)

The edit distance with real penalty (EDR) [55]  $d_{ed11}$  is a variation of the native edit distance and is applicable to real-valued sequences. Two values are equal if they are within a given threshold,  $\delta$ .

$$d_{ed11}(x_i, x'_j) = \min \begin{cases} 0\ if\ |x_i - x'_j| \leq \delta \\ 1\ if\ x_i\ or\ x'_j\ is\ a\ gap \\ 1\ otherwise \end{cases} \quad (4.11)$$

## 4.2.2 Example derivation of edit distance

In this subsection, we illustrate with examples how the edit distance is derived between a pair of patients. Data for two patients (patient 1 and patient 2) is extracted from the clinical table. See Table B.1 for a sample of a few examples of raw input data from the clinical table. The data is displayed in sequential order according to the date the events were recorded. Table 4.1 is an example of data extracted for patient 1, while Table 4.2 represents data for patient 2. The recorded event date, the read code of the event record, and its corresponding read term are extracted. Events recorded in the clinical table include clinical events such as procedures, conditions, and symptoms.

The read codes are extracted and represented as a sequence of symbols, see Figure 4.5. As can be seen, the length of both sequences is not the same. Patient 1 has more items than patient 2, as would be the case in an entire population. The next goal of the process is to quantify the dissimilarity between Patient 1 and patient 2 using edit distance. The read codes from Table 4.1 and Table 4.2 are treated as a sequence of symbols in Figure 4.5. The symbols are aligned, and where they differ, we substitute the symbol for patient 1 with the symbol for patient 2 at a cost of 2. Where the symbols are similar, we do nothing. Where there is a gap for the corresponding item in patient 2, we delete the items for patient 1. The cost of one is applied to the deletion. The total cost is calculated and summed up. Computing the edit distance can be best achieved using dynamic programming. An example of a tabular array used in the dynamic programming process is displayed in Figure 4.3. The total distance between both sequences is calculated as 12.

<i>Event Date</i>	<i>Read Code</i>	<i>Read Term</i>
19890807	AB0..00	TINEA
19890807	1371.00	Never smoked tobacco
19890807	1362.11	Drinks rarely
19890807	229..00	O/E - height
19890807	22A..00	O/E - weight
19890807	246..00	O/E - blood pressure reading
19891224	M03z.00	CELLULITIS
19921020	4K4Z.00	Cervical smear action NOS
19930928	S8z..13	LACERATION
19940703	9OW..00	New patient screen admin.

Table 4.1: Table showing sample data extracted from the clinical table for patient 1

<i>Event date</i>	<i>Read Code</i>	<i>Read Term</i>
19951024	C10..00	Diabetes mellitus
19951024	137..00	Tobacco consumption
19951024	136..00	Alcohol consumption
19951206	229..00	O/E - height
19951206	22A..00	O/E - weight
19960309	246..00	O/E - blood pressure reading
19960309	115..00	No significant medical history
19970814	1225.11	No FH: CVA/Stroke/TIA

Table 4.2: Table showing sample data extracted from the clinical table for patient 2

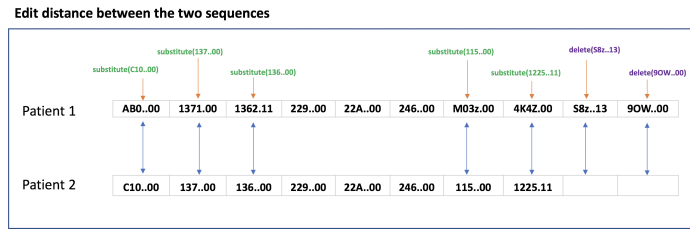


Figure 4.3: Edit distance applied to both sequence of symbols

		AB0..00	1371	1362.11	229..00	22A..00	246..00	M03z.00	4K4Z.00	S8z..13	9OW..00
	0	1	2	3	4	5	6	7	8	9	10
C10..00	1	2	3	4	5	6	7	8	9	10	11
137..00	2	3	4	5	6	7	8	9	10	11	12
136..00	3	4	5	6	7	8	9	10	11	12	13
229..00	4	5	6	7	6	7	8	9	10	11	12
22A..00	5	6	7	8	7	6	7	8	9	10	11
246..00	6	7	8	9	8	7	6	7	8	9	10
115..00	7	8	9	10	9	8	7	8	9	10	11
1225.11	8	9	10	11	10	9	8	9	10	11	12

Figure 4.4: Dynamic programming approach used to calculate the Edit distance of both sequences

### 4.3 Proposed Kernels

Via the so-called kernel trick, a kernel is equivalent to an implicit mapping of entity pairs into a high-dimensional feature space, followed by a inner product in that space. It is thus a symmetric function  $K : X \times X \mapsto \mathbb{R}$  such that,

$$\forall x_i, x_j \in X, k(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle \tag{4.12}$$

Where  $\phi : X \mapsto F$  is a function map  $\phi$  that converts the input  $X$  into a high dimensional feature space  $F$ . The notation  $x_i$  used in this paper corresponds to the sequence of symbols encoding clinical interventions, symptoms, diagnosis, procedures, and medication for a single patient. If a valid kernel



function meets the following condition, it is positive definite.

$$k(x_i, x_j) = \sum_{i,j=1}^n c_i c_j k(x_i, x_j) \geq 0 \quad (4.13)$$

for any  $x_1, \dots, x_n \in X$  and  $c_1, \dots, c_n \in \mathbb{R}$ , or, equivalently, that its matrix's eigenvalues are all non-negative. See Figure 4.2 for the conceptual framework.

### 4.3.1 Proposed kernel construction methods

In this section, we describe various techniques that are used in developing bespoke kernel functions in conjunction with variants of edit distances. Given a pair of symbolic sequences  $\mathbf{x} = x_1, x_2, \dots, x_n$  and  $\mathbf{x}' = x'_1, x'_2, \dots, x'_m$  i.e  $\mathbf{x}, \mathbf{x}' \in \{\mathbf{X}\}_{i=1}^m$  of data derived from Alphabet  $\Sigma$ , the kernel functions on the pair of sequences are described below:

#### 1. Edit Pseudo kernel functions

As pseudo kernel functions, we use the distance measure directly. Doing so contravenes the definition of a kernel function as a similarity measure. Nevertheless, the kernel learners used in the experiments are adaptable to learning with pseudo kernels.

$$K(\mathbf{x}, \mathbf{x}') = d(.,.) \quad (4.14)$$

where  $d(.,.)$  is the edit distance between a pair of sequences. This forms the baseline for the experiments

**Example:**

$$K_{ed1}(\mathbf{x}, \mathbf{x}') = d_{ed1}(\mathbf{x}, \mathbf{x}') \quad (4.15)$$

The formulation in 4.15 constructs a pseudo kernel function using the native edit distance  $d_{ed1}$  defined in section 4.2.1. The evaluation of 4.15 on the EHR data yields a matrix with zeros in its diagonal; thus, these are not really valid kernels. Nevertheless, we apply these directly as input pseudo-kernels into the SVM classifier. All defined edit distances from section 4.2.1 are implemented as pseudo kernels, which serve as the baseline for comparison against the performance achieved from other kernel construction methods (see table 4.3 for the comprehensive list of kernels).

#### 2. Conversion to similarity measure

Several methods can be used to convert a distance to a similarity-proximity measure. Both measures apply a real value that quantifies the closeness or disparity between objects. Unlike distance functions, which assign a small value to identical objects and a large value to dissimilar objects, similarity measures do the opposite. Similar items are assigned larger values than dissimilar items.

$$K(\mathbf{x}, \mathbf{x}') = \frac{1}{1 + d(.,.)} \quad (4.16)$$

where  $\frac{1}{1+d(.,.)}$  converts the distance  $d(.,.)$  into a similarity measure.

**Example:** The similarity conversion method converts the edit distance  $d_{ed1}$  into a similarity measure and, as a result, potentially a valid kernel (see table 4.3 for the comprehensive list of kernels).

$$K_{s.ed1}(\mathbf{x}, \mathbf{x}') = \frac{1}{1 + d_{ed1}(\mathbf{x}, \mathbf{x}')} \quad (4.17)$$

### 3. Gaussian Edit kernel

We can use the Gaussian RBF function approach to convert distances into valid kernels. The traditional Euclidean distance is replaced by the computed edit distance.

$$K(\mathbf{x}, \mathbf{x}') = e^{-\gamma \cdot d(\cdot, \cdot)} \quad (4.18)$$

Where the positive parameter  $\gamma$  scales the kernel for numerical stability. At certain values,  $\gamma$  makes the kernel psd. Choosing  $\gamma$  is usually accomplished through the costly cross-validation method. The choice of parameter value can also affect the classification outcome. It overfits the data if it is too small. When it is too large, it can generate a diagonally dominant kernel matrix, which, while ensuring PSD, does not generalize well.

**Example:**

$$K_{G\_ed1}(\mathbf{x}, \mathbf{x}') = e^{-\gamma \cdot d_{ed1}(\mathbf{x}, \mathbf{x}')} \quad (4.19)$$

(See table 4.3 for the comprehensive list of kernels).

### 4. Rational Quadratic Edit kernel

The rational quadratic kernel is modified in a similar manner as the Gaussian edit kernel by substituting the traditionally Euclidean distance with the edit distance.

$$K(\mathbf{x}, \mathbf{x}') = 1 - \frac{d(\cdot, \cdot)}{d(\cdot, \cdot) + c} \quad (4.20)$$

where  $c \in \mathbb{R}^+$

**Example:**

$$K_{r\_ed1}(\mathbf{x}, \mathbf{x}') = 1 - \frac{d_{ed1}(\mathbf{x}, \mathbf{x}')}{d_{ed1}(\mathbf{x}, \mathbf{x}') + c} \quad (4.21)$$

(See table 4.3 for the comprehensive list of kernels).

### 5. Polynomial Edit kernel

The polynomial kernel function is one of the popular kernel functions that is applicable to vectorized data. It is a polynomial implementation of the linear (dot product) kernel, with the free parameter  $c$  controlling the influence of higher-order polynomial terms on lower-order terms. In this case, the value is set to 1. The dot product in a traditional polynomial kernel is replaced with the similarity measure derived from the edit distance. This guarantees that we are working with a similarity measure..

$$K(\mathbf{x}, \mathbf{x}') = (d(\cdot, \cdot) + c)^\alpha \quad (4.22)$$

**Example:**

Traditional polynomial kernels that are applied to vector spaces are implemented with the inner products of the examples. In this case, we replace  $d(\cdot, \cdot)$  with the edit distance similarity conversion derived from two symbolic sequences.

$$K_{p\_ed1}(\mathbf{x}, \mathbf{x}') = (K_{s\_ed1}(\mathbf{x}, \mathbf{x}') + c)^\alpha \quad (4.23)$$

(See table 4.3 for the comprehensive list of kernels).

**Justification:** This is the *generalized polynomial* kernel, which can be built on top of any other valid base kernel as in  $k(x, z) = p(k_0(x, z))$ , where the base kernel  $k_0$  is a valid kernel (in this case the similarity conversion of the edit distance makes it a valid kernel) and  $p : \mathbb{R} \mapsto \mathbb{R}$  is a polynomial function with non-negative coefficients [85].

## 6. Edit Distance Substitution kernel

Distance measures are metrics that obey the triangular inequality. They generate non-negative values and have zeros along the diagonals of their symmetric matrix. Since PSD kernels are generalizations of vector products in the induced Mercer feature space, we can extend the concept of PSD kernels to a larger class of kernels known as conditionally positive kernels (cpd), expressed in terms of the norms of the embedding feature space. Thus, the kernel function can be used to express the norm  $\|\phi(x_i) - \phi(x_j)\|^2$ , which quantifies how close objects are in the feature space:

$$\|\phi(x_i) - \phi(x_j)\|^2 = k(x_i, x_i) + k(x_j, x_j) - 2k(x_i, x_j) \quad (4.24)$$

where  $k(., .)$  denotes a kernel function.

As a result, we are able to apply distance metrics, in this case edit distance, in the construction of kernels. A distance measure is said to be isometric to the L2-norm if the data can be embedded in a Hilbert space such that  $d(x, x_0) = \|\phi(x) - \phi(x_0)\|$  (this approach is termed a ‘distance substitution kernel’).

In contrast to dot products, norms are invariant to translations, so  $x \mapsto x_i - x_0$ . The dot product of the translation can be expressed as

$$\langle (x_i - x_0), (x_j - x_0) \rangle = \frac{1}{2} \left( -\underbrace{\|x_i - x_j\|^2}_{d(x_i, x_j)} + \underbrace{\|x_i - x_0\|^2}_{d(x_i, x_0)} + \underbrace{\|x_0 - x_j\|^2}_{d(x_0, x_j)} \right) \quad (4.25)$$

For any  $x_0 \in X$  we show this to be a valid PSD kernel by

$$\sum_{i,j} c_i, c_j \langle (x_i - x_0), (x_j - x_0) \rangle = \sum_{i,j} c_i \| (x_i - x_0) \|^2 \geq 0 \quad (4.26)$$

A conditionally positive definite symmetric  $n \times n$  matrix  $K$  ( $m \geq 2$ ), on the other hand, satisfies the condition in equation 4.13 for any  $x, \dots, x_n \in X$  and  $c, \dots, c_n \in \mathbb{R}$ , but with the added property  $\sum_{i=1}^n c_i = 0$ .

### Example:

We compute the edit distance from pattern  $x$  to  $x_0$  and from  $x_0$  to  $x'$  in relation to the distance from  $x$  to  $x'$ , as detailed in equation 4.25.

$$K_1(x, x') = \frac{1}{2} (d(x, x_0)^2 + d(x_0, x')^2 - d(x, x')^2) \quad (4.27)$$

where  $d(., .)^2$  is substituted by the edit distance  $d_{ed1}(x, x_0)$  of two symbols. We can therefore re-write equation 4.27.

$$K_{ds\_ed1}(x, x') = \frac{1}{2} (d_{ed1}(x, x_0) + d_{ed1}(x_0, x') - d_{ed1}(x, x')) \quad (4.28)$$

(See table 4.3 for the comprehensive list of kernels).

## 7. Edit distance Template Matching kernel

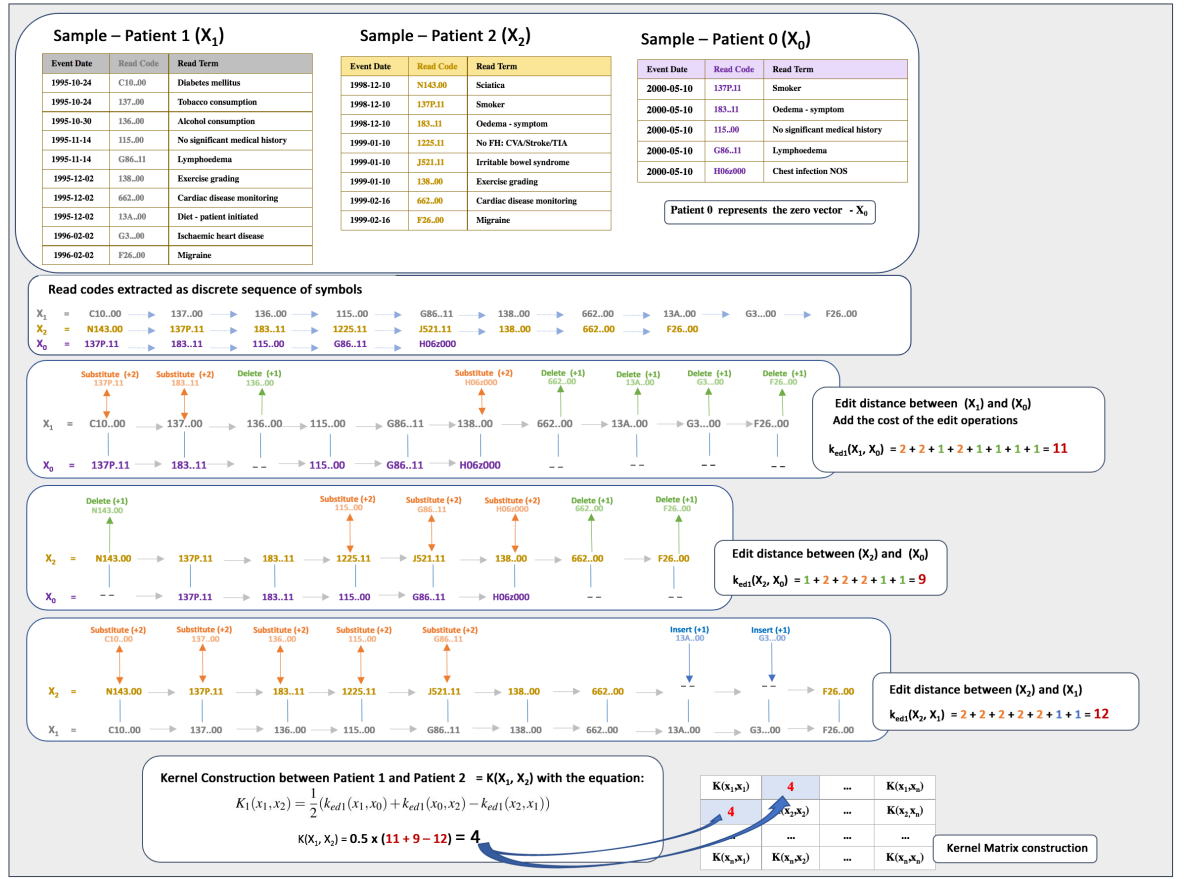


Figure 4.5: Evaluating the pairwise kernel function by first extracting the data as a sequence of symbols, then computing the edit distance between a pair of sequences. An edit cost of 2 is applied where symbols are **substituted** while 1 is applied if a symbol is **deleted** or **inserted**. The total cost is computed and used to derive the kernel value as specified in Equation 4.27

Similar to the template matching method, a kernel function between a pair of sequences is computed by applying edit distance between each sequence and a template before computing their product.

$$K(\mathbf{x}, \mathbf{x}') = (d(x, x_0) * d(x_0, x')) \quad (4.29)$$

where  $x_0 \in \mathbf{X}$  is the template sequence selected from the training example. All data points  $\mathbf{x}$  are potential candidates for the template sequence.

**Example:** Because this kernel necessitates a search for the best candidate to serve as the template sequence, we have only implemented it on edit distance  $d_{edit}$ .

$$K(\mathbf{x}, \mathbf{x}') = (d_{edit}(x, x_0) * d_{edit}(x_0, x')) \quad (4.30)$$

### 4.3.2 Spectral modification

Learning with a positive semidefinite kernel is necessary for the SVM algorithm's convergence to a solution. Nevertheless, it is still possible to learn directly from an indefinite kernel and obtain good classification results. Solutions from indefinite kernels were previously hard to interpret due to a lack of geometrical and theoretical understanding [111]. The following spectral modifications [155] can be applied to convert the negative eigenvalues to PSD.

1. **Clip** The negative eigenvalue is simply removed

ED	Description	Pseu	Sim	Gaus	RQ	Poly	DS	TM
$d_{ed1}$	Edit distance	$k_{ed1}$	$k_{s\_ed1}$	$k_{G\_ed1}$	$k_{r\_ed1}$	$k_{p\_ed1}$	$k_{ds\_ed1}$	$k_{t\_ed1}$
$d_{ed2}$	Edit distance with length normalization	$k_{ed2}$	$k_{s\_ed2}$	$k_{G\_ed2}$	$k_{r\_ed2}$	$k_{p\_ed2}$	$k_{ds\_ed2}$	
$d_{ed3}$	Edit distance normalized by the number of common items	$k_{ed3}$	$k_{s\_ed3}$	$k_{G\_ed3}$	$k_{r\_ed3}$	$k_{p\_ed3}$	$k_{ds\_ed3}$	
$d_{ed4}$	Edit distance normalized by the exponent of the number of common items	$k_{ed4}$	$k_{s\_ed4}$	$k_{G\_ed4}$	$k_{r\_ed4}$	$k_{p\_ed4}$	$k_{ds\_ed4}$	
$d_{ed5}$	Subsequence edit distance	$k_{ed5}$	$k_{s\_ed5}$	$k_{G\_ed5}$	$k_{r\_ed5}$	$k_{p\_ed5}$		
$d_{ed6}$	Subsequence edit distance normalised by edit distance between the pair of sequences	$k_{ed6}$	$k_{s\_ed6}$	$k_{G\_ed6}$	$k_{r\_ed6}$	$k_{p\_ed6}$		
$d_{ed7}$	Subsequence edit distance normalised by the number of common symbols	$k_{ed7}$	$k_{s\_ed7}$	$k_{G\_ed7}$	$k_{r\_ed7}$	$k_{p\_ed7}$		
$d_{ed8}$	Subsequence edit distance normalised by the number of items common to each pair of subsequences	$k_{ed8}$	$k_{s\_ed8}$	$k_{G\_ed8}$	$k_{r\_ed8}$	$k_{p\_ed8}$		
$d_{ed9}$	Edit distance computed with number unmatched and matched symbols	$k_{ed9}$	$k_{s\_ed9}$	$k_{G\_ed9}$	$k_{r\_ed9}$	$k_{p\_ed9}$		
$d_{ed10}$	Edit distance with with controlled equality	$k_{ed10}$						
$d_{ed11}$	Edit distance on real sequence	$k_{ed11}$						

Table 4.3: Edit kernel functions constructed with the formulations expressed in equations 4.14, 4.16, 4.18, 4.20, 4.22, and 4.24 in conjunction with variants of edit distances defined in section 4.2.1. Where **ED**: Edit Distance, **Pseu**: pseudo, **Sim**: Similarity, **Gaus**: Gaussian, **RQ**: Rational Quadratic, **Poly**: Polynomial, **DS**: Distance Substitution, **TM**: Template Matching

2. **Shift** The complete spectrum is shifted until the least eigenvalue is 0
3. **Flipping** - The absolute value of the spectrum is used
4. **Square** - The eigenvalues are squared, which is equivalent to squaring the kernel matrix  $X^T X$

## 4.4 Kernel-based Learning Algorithms

### 4.4.1 Support Vector Machine

The support vector machine (SVM) is a deterministic classifier that aims to find the optimum separating hyperplane that can split data into two distinct classes. It finds the largest margin between the two classes. However, most real-life problems are not linearly separable, and where they are, a greater margin can only be achieved if some errors are allowed. Thus, a soft margin classifier that allows some errors can be formulated as the following function to be optimized:

$$\min_{\omega, b, \xi} \frac{1}{2} \omega^T \omega + C \sum_{i=1}^l \xi_i$$

subject to

$$y_i(\omega^T \phi(x_i) + b) \geq 1 - \xi_i$$

$$\xi_i \geq 0, \quad i = 1, \dots, l$$

where  $\phi(x_i)$  maps  $x_i$  into a higher-dimensional space and  $C > 0$ , where  $C$  is the regularization parameter that the user selects. It allows us control the trade-off between a large margin and classification error. We obtain the dual formulation of the optimization problem expressed in terms of Lagrange multipliers  $\alpha$  as:

$$\max_{\alpha} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n y_i y_j \alpha_i \alpha_j \langle x_i, x_j \rangle$$

subject to:

$$\sum_{i=1}^n y_i \alpha_i = 0, \quad 0 \leq \alpha_i \leq C$$

The weight vector  $\mathbf{w}$  can be written in terms of the examples  $x_i$  and the solution  $\alpha_i$  of the optimization problem as

$$\omega = \sum_{i=1}^l y_i \alpha_i \phi(x_i)$$

The  $x_i$  values for which  $\alpha_i > 0$  are referred to as **support vectors**. These are the only data points that contribute to determining the margin. The dual formulation of the problem expressed in terms of the dot product of the data points,  $\langle x_i, x_j \rangle$ , can be replaced by a positive semidefinite kernel function,  $k(x, x)$ . This makes SVM algorithms applicable to all types of data.

The discriminatory decision function is given as

$$\text{sgn}(\omega^T \phi(x) + b) = \text{sgn} \left( \sum_{i=1}^l y_i \alpha_i K(x_i, x) + b \right)$$

The LibSVM [51] algorithm will be applied as the SVM classification algorithm.

#### 4.4.2 Gaussian Process

Gaussian process (GP) is a generalisation of a probability distribution to functions. It is a collection of random variables for which a finite collection has a joint Gaussian distribution [203]. It is defined by the mean function  $m(\mathbf{x})$  and the covariance function  $\Sigma(\mathbf{x})$ , which is given by  $\Sigma_{i,j} = K(x_i, x_j)$ , where  $K$  is a positive semidefinite kernel or covariance function. A Gaussian process defines a prior over a set of functions. Therefore, it can be used for inference by estimating the posterior mean function once we've seen some data. A process  $f(x)$  defined as a GP is given as

$$f(x) \sim \mathcal{GP}(m(x), \Sigma(x))$$

Given a dataset  $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ , our goal in regression is to estimate a function  $y = f(x) + \epsilon$  where  $\epsilon \sim \mathcal{N}(0, \sigma^2)$  is regarded as noise. We infer a distribution over functions given the data,  $p(f(x)|\mathbf{X}, \mathbf{y})$ , and use this to make predictions when presented with a new example  $x_*$  by computing

$$p(y_*|x_*, \mathbf{X}, \mathbf{y}) = \int p(y_*|f, x_*) p(f|\mathbf{X}, \mathbf{y}) df$$

The Gaussian prior distribution over the observed target  $f(x)$  specified by 0 mean and covariance function  $K(x_i, x_j)$  is expressed as

$$f(\mathbf{x}) \sim \mathcal{N}(0, K(\mathbf{X}, \mathbf{X}') + \sigma^2 I)$$

Using the Gaussian process approach, we can estimate  $\mathbf{f}_*$  for an unseen example  $x_*$ , by computing

the joint distribution over the observed  $\mathbf{f}$  and the new example  $x_*$  using

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N} \left( \begin{bmatrix} \mu \\ \mu_* \end{bmatrix}, \begin{bmatrix} K(X, X) + \sigma_n^2 I & K(X, x_*) \\ K(x_*, X) & K(x_*, x_*) \end{bmatrix} \right)$$

$\mathbf{f}$  represents the function values of the training cases and  $\mathbf{f}_*$  represents the function values of the test cases  $X_*$ .

## 4.5 Multi-Kernel Learning (SimpleMKL)

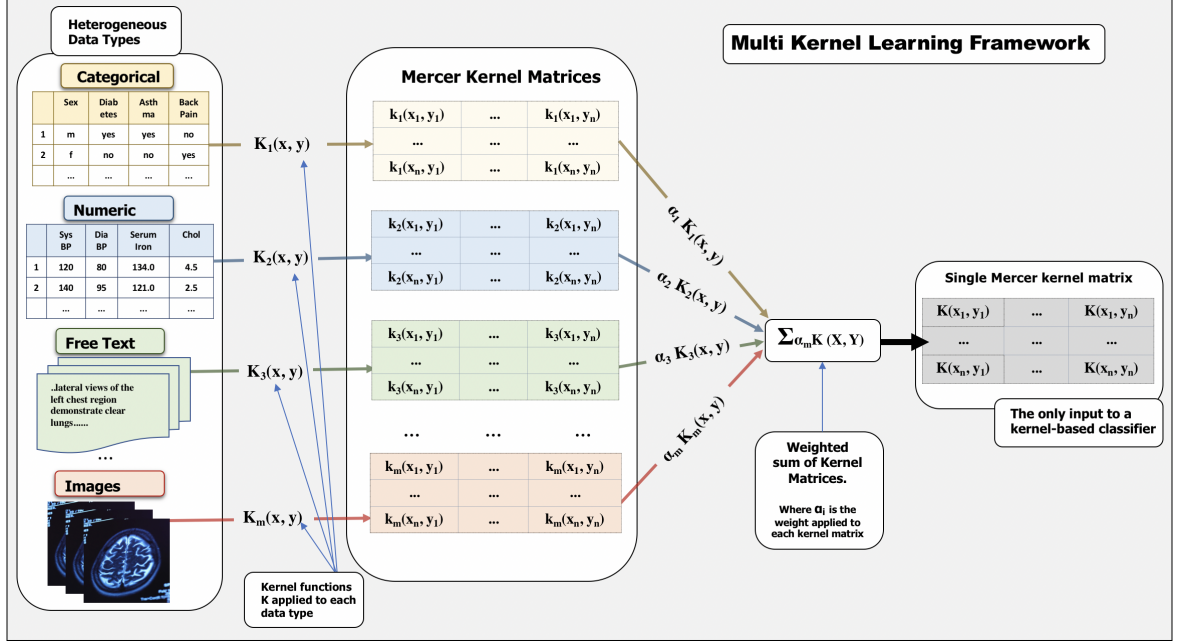


Figure 4.6: The multi-kernel learning (MKL) framework for combining kernels derived from disparate data types

Several methods have been proposed and used to combine base kernels. For the purposes of this experiment, we describe and employ the SimpleMKL [202]. The objective is to optimise the combination of  $n$  basis kernels  $K_m(x, y)$ , ( $m = 1, \dots, m$ ) and an SVM classifier. Therefore, with SimpleMKL, we can achieve this by optimizing:

$$K(x, y) = \sum_{m=1}^M d_m K_m(x, y)$$

subject to  $d_m \geq 0$ , and

$$\sum_{m=1}^M d_m = 1$$

where  $d_m$  denotes the weight of the kernels. In addition to solving an SVM classifier:

$$f(x) = \sum_{i=1}^l \alpha^* K(x, x_i) + b^*$$

$b^*$  and  $\alpha^*$  are the SVM coefficients to be learned. Therefore, the goal of the MKL solver is to learn both SVM coefficients and combination weights simultaneously. This can be achieved by solving:

$$\begin{aligned}
 & \min \frac{1}{2} \left\| \sum_i \alpha_i^* K(\cdot, x_i) \right\|^2 + C \sum_i \xi \\
 & \text{s.t. } y_i \sum_i \alpha_i K(x_i, x_j) + y_i b \geq 1 - \xi_i \\
 & \quad \xi_i \geq 0 \\
 & \quad \sum_{m=1}^M d_m = 1, d_m \geq 0
 \end{aligned}$$

We refer the reader to the paper [202], where the solution to the SimpleMKL algorithm is covered in detail.

## 4.6 Kernel Evaluation and Selection

The methods described in this section were used to evaluate the appropriateness of kernel functions. These were applied to establish, rank, and assess the quality of kernels.

### 4.6.1 Kernel Alignment

Kernel alignment [64] which can measure the similarity between two kernel functions or between a kernel and a target, was used to evaluate the quality of the kernels developed. The kernel alignment of  $K_1$  and  $K_2$  is given as,

$$A(K_1, K_2) = \frac{\langle K_1, K_2 \rangle}{\sqrt{\langle K_1, K_1 \rangle \langle K_2, K_2 \rangle}}$$

This can be viewed as the cosine of the angle between two bi-dimensional vectors,  $K_1$  and  $K_2$ .

If we consider  $K_2 = yy'$  for a given vector  $y$  of labels  $\{+1, -1\}$ , the kernel target alignment is given as,

$$A(K_1, yy') = \frac{\langle K_1, yy' \rangle}{\sqrt{\langle K_1, K_1 \rangle \langle yy', yy' \rangle}}$$

It is the (normalized) Frobenius inner product between the kernel matrix and the covariance matrix of the target vector. This quantity captures the degree of agreement between a kernel and a given learning task [64].

### 4.6.2 Spectral Ratio

The spectral ratio used in evaluating the expressiveness of kernels in a multi-kernel learning (MKL) framework [85] was also employed in this study. The expressiveness of a kernel function is the number of dichotomies that can be realized by a linear separator in that feature space [85]. It is defined as the ratio between the 1-norm and 2-norm of the kernel eigenvalues. It is equal to the ratio between the trace norm  $\|K\|_T$  and its Frobenius norm  $\|K\|_F$ .

$$C(K) = \frac{\sum_{i=1}^L \lambda_i}{\sqrt{\sum_{i=1}^L \lambda_i^2}} = \frac{\|K\|_T}{\|K\|_F} = \frac{\sum K_{ii}}{\sqrt{\sum_{ij} K_{ij}^2}}$$



Given two kernel functions  $K_1$  and  $K_2$  derived on dataset  $X$ , then  $K_1$  is more general (or less expressive) than  $K_2$  if the spectral ratio computed on both kernels gives  $C(K_1) \leq C(K_2)$ . This means  $K_2$  is more specific (or more expressive) than  $K_1$ .

### 4.6.3 Classification performance and evaluation

Each kernel was developed in conjunction with the SVM algorithm (LibSVM, [51]). A 5-fold cross-validation was initially proposed for this experiment. However, testing the effect of variable-length sequences on classification performance reduced the number of examples; therefore, the leave-one-out cross validation (LOOCV) method was used. This is typically appropriate when learning from small amounts of data and should provide a more accurate estimate of generalization performance.

The following performance metrics were collected for the predicted labels: true positive TP, true negative TN, false negative FN, and false positive FP.

#### Accuracy

The classification accuracy is given as the ratio of the correct positive and negative predicted outcomes to all predicted outcomes.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

This, however, does not give the true reflection of performance if there is a class imbalance from an uneven distribution of positive and negative outcomes.

#### Precision

Precision measures the ratio of the correctly predicted positive labels to the total number of all predicted positive labels,

$$\frac{TP}{TP + FP}$$

#### Recall

Recall (sensitivity) measures the ratio of the correctly predicted positive labels to all the actual positive labels,

$$\frac{TP}{TP + FN}$$

#### F1-score

The F1-score is derived by calculating the weighted average of precision and recall.

$$F_1Score = 2 * \frac{Recall * Precision}{Recall + Precision}$$

The F1-score gives a better evaluation of the model's performance since it takes into account the wrongly predicted positive and negative labels. The higher the F1-score, the better.

## 4.7 Baseline Models

The proposed framework is compared against established classical methods used in feature representation and classification of sequential data, such as in natural language processing (NLP) tasks.

The bag-of-words (BoW) method of extracting even-length numeric feature vectors is adopted for this purpose. Bag-of-words yields a histogram of data entities representing the frequency of occurrence for each patient. The tabular data matrix contains 3054 distinct clinical codes, with one code per column. This was extracted from the (all data) dataset. In contrast, the binary bag-of-words feature representation encodes the presence or absence of the 3054 distinct clinical codes for each patient record.

The following classification algorithms—logistic regression, SVM, and deep learning recurrent neural networks (RNNs) with long short-term memory (LSTM)—are used to classify the data. These are applied as the baseline to evaluate the classification performance and suitability of the proposed method.

### 4.7.1 Baseline kernel functions

As baseline models, the following base kernel functions were used with SVM:

- **Linear** The linear kernel is the dot product similarity measure and one of the basic kernels used with SVM for linearly separable data.

$$K(x, y) = (x^\top y) \quad (4.31)$$

- **Polynomial kernel** The polynomial kernel is defined as

$$K(x, y) = (x^\top y + c)^d \quad (4.32)$$

where  $d$  is the degree of the polynomial and  $c \geq 0$  is a free parameter that controls the influence of higher order terms of the polynomial.

- **Radial Basis Function (RBF) kernel** The radial basis function (RBF) kernel, also referred to as the Gaussian kernel, is well suited for numeric data. It has some interesting properties that make it suitable for a lot of classification tasks. Its free parameter  $\frac{1}{2\sigma}$  can be used to control the performance of the kernel.

$$K(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma}\right) \quad (4.33)$$

- **Exponential RBF kernel** The exponential RBF kernel differs from the Gaussian RBF kernel by its norm which is not squared.

$$K(x, y) = \exp\left(-\frac{\|x - y\|}{2\sigma}\right) \quad (4.34)$$

- **Laplace kernel** This kernel function is also a part of the RBF family of kernels. It is similar to the exponential RBF except that it is not too sensitive to its free parameter  $\sigma$

$$K(x, y) = \exp\left(-\frac{\|x - y\|}{\sigma}\right) \quad (4.35)$$

## 4.8 Dataset

Experiments are conducted using anonymised test dummy primary health care data that reflects actual medical data distribution. It is modelled after Vision 3 General Practice IT System data. We

searched the database for patients with the read code C10..00 for diabetes mellitus. 158 out of 9628 patients that met the inclusion criteria were used for this study. Each patient record was checked for the presence of a systolic BP of 130 mm Hg and a diastolic BP of 80 mm Hg recorded prior to being diagnosed with type 2 diabetes. The presence or absence of elevated blood pressure was used as the outcome variable. Those with prior blood pressure readings of 130/80 mm Hg or higher were labelled as positive, while those with readings less than 130/80 mm Hg were labelled as negative. 42% (66) of the 158 patients were labelled as positive, while 58% (92) were labelled as negative. The self-identified gender distribution is 76 females, 82 males. The youngest patients at the index cut-off date (02/13/2002, the last date an item was recorded in the database) are 31 years old, while the oldest are 107 years old. The average age is 71 years (95% CI 69.28 - 74.23).

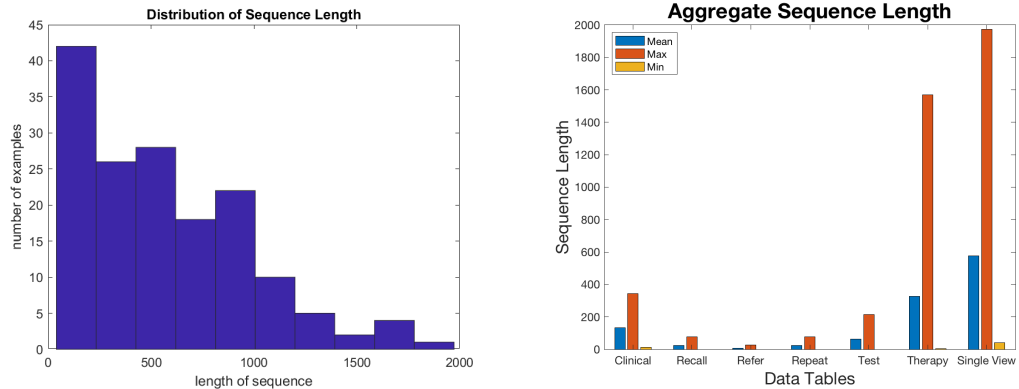


Figure 4.7: Plots showing the uneven sequence length distribution and the aggregate mean, maximum and minimum length distribution according to the datasets

Table	Description	min	max	Mean $\pm$ (std)
<b>Clinical</b>	Patient Clinical Data (Medical Histories)	20	347	138 (84)
<b>Recall</b>	Recalls, typically for immunisations, cytology etc. (reminders for the patient to return to the surgery for continuing treatment)	0	77	23 (18)
<b>Refer</b>	Referrals to third parties (Consultants)	0	27	6 (5)
<b>Repeat</b>	Repeat Therapy items	0	77	23 (17)
<b>Test</b>	Clinical test results	0	215	63 (59)
<b>Therapy</b>	Acute (one off) and Repeat Issue therapy items	4	1570	327 (306)
<b>All</b>	Symbolic data from all 6 table extracted as a single dataset	45	1976	579 (419)

Table 4.4: Details of the data tables showing their description, minimum, maximum, and mean length of the sequences

Each patient record consists of a discrete symbolic sequence of between 40 and 1974 data items. The distribution of the variable-length sequences is displayed in Figure 4.7. Table 4.1 shows a sample of data extracted for a single patient (see Appendix B for more examples of the EHR data). The read codes, ordered by the event dates, are extracted as discrete symbols.

#### 4.8.1 Data preprocessing/cleansing

Records with missing values were ignored in the extract.

1. **Clinical Table:** This data table has the journal entries;
  - 43 events without a read code and read term were deleted

<i>Event Date</i>	<i>Read Code</i>	<i>Read Term</i>
19890807	AB0..00	TINEA
19890807	1371.00	Never smoked tobacco
19890807	1362.11	Drinks rarely
19890807	229..00	O/E - height
19890807	22A..00	O/E - weight
19890807	246..00	O/E - blood pressure reading
19891224	M03z.00	CELLULITIS
19921020	4K4Z.00	Cervical smear action NOS
19930928	S8z..13	LACERATION
19940703	9OW..00	New patient screen admin.

Table 4.5: Table showing sample data extracted from the clinical table for patient 1

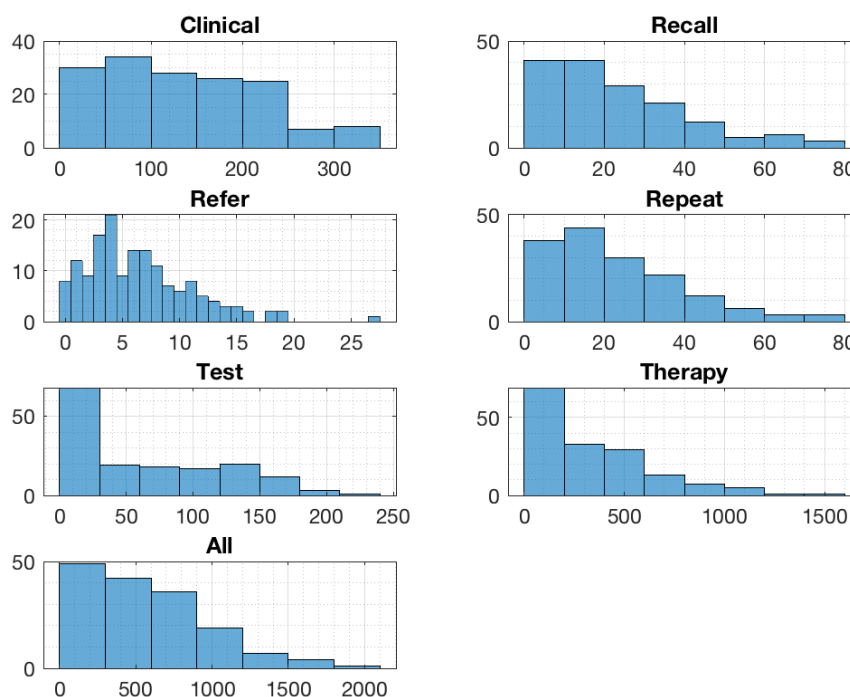


Figure 4.8: Plot showing distribution of length of sequences per data table

- The date format was changed to `yyyymmdd`
  - Deleting outliers
    - Example 102 had an item (“stopped smoking”) recorded 19290101 with a gap of 44 years to the next item recorded ‘19730101’
    - Example 30 had an item (“stopped smoking”) recorded 19300101 with a gap of 30 years to the next item recorded ‘19600101’
    - Example 136 had an item recorded 19010701 with a gap of 39 years to the next item recorded “19390101”
  - Dates with the year but missing either month and day were set to year 01 01
  - Read Code, Read Term, event dates were extracted
2. **Recall Table:** This data table contains the recalls. It contains the read code and read term, ordered by the event date. No further preprocessing was done. The read code, read term, and event date were extracted

	<b>Read Term</b>	<b>Read Code</b>	<b>Description</b>	<b>Mean (std) Values</b>
1	Serum cholesterol	44P..00	The serum cholesterol level can indicate the risk of developing heart disease or stroke. Normal levels are expected to be within 5.5 mmol/L (micromoles/litre)	5.73 (5.04)
2	Serum creatinine	44J3.00	Measures how well the kidneys filters waste from the blood. Normal range for adult men is 65.4 to 119.3 mmol/L (micromoles/litre) and women, 52.2 to 91.9 mmol/L	90.68 (31.31)
3	Serum sodium	44I5.00	Measures the concentration of sodium in the blood. A measure to determine likelihood of Hyponatremia. Normal blood sodium level is between 135 and 145 mEq/L (milliequivalents/litre) and Hyponatremia occurs when this falls below 135 mEq/L	139.95 (2.89)
4	Serum potassium	44I4.00	Measures potassium levels in blood. Normal serum potassium is 3.5 to 5.5 mEq/L (milliequivalents/litre). Hyperkalemia occurs when the potassium levels are high. Heart attack occurs at extremely high levels.	4.38 (0.49)
5	Serum albumin	44M4.00	An indicator that measures the oncotic pressure needed for proper distribution of body fluids. Normal value for serum albumin in blood is 3.4 to 5.4 grams per % deciliter. It can lead to Hypoalbuminemia	38.68 (3.06)
6	Serum alkaline phosphatase	44F..00	Measures the amount of alkaline phosphatase (ALP) in the blood. High levels of ALP can indicate liver disease or bone disorder. Recommended normal range is 30 to 120 IU/L (international units/litre)	91.77 (44.34)
7	Serum bilirubin level	44E..00	Measures the level of bilirubin in the blood. An indication of how healthy the liver is. Normal total bilirubin 1.71 to 20.5 $\mu$ mol/L (micromole/litre)	11.11 (4.26)
8	ALT/SGPT serum level	44G3.00	Measures the level of alanine aminotransferase. Normal levels of ALT (SGPT) ranges from about 7-56 (units/litre) of serum. High levels indicate liver disease.	34.20 (20.09)
9	Corrected serum calcium level	44IC.00	Measures total calcium concentration. Normally 2.2-2.6 mmol/L. Level can indicate hypocalcemia or hypercalcemia	2.30 (0.10)
10	AST - aspartate transam.(SGOT)	44H5.00	The Aspartate aminotransferase (AST) test checks for liver damage. Normal range for adult men 14–20 U/L and women 10–36 U/L (units/litre)	28.18 (22.00)
11	Blood glucose result	44U..00	An indicator for diabetes. A blood sugar level less than 7.8 mmol/L (millimoles/litre) is normal while 11.1 mmol/L after two hours indicates diabetes, between 7.8 mmol/L and 11.0 mmol/L indicates pre diabetes	9.93 (4.52)

Table 4.6: Table showing 11 Test Events from the Test dataset with numeric values. The Test Read code, Read term, description showing the normal range, unit of measure, and the mean (std) values extracted from the dataset are displayed

- Refer Table:** The Refer table stores information about the referrals that were sent. The referral read code, read term, and event dates are extracted. Null or missing records are excluded.
- Repeat Table:** The Repeat table contains repeat medication orders made. The drug name,

dosage, term, form, and event date were extracted.

5. **Test Table:** This data table holds the test results for the examples. It contains various tests and measurements. 52 unique test entities, of which 30 contain numeric values. The event date, test read codes, read term, and any corresponding values are extracted. Null or missing records are excluded.
6. **Therapy Table:** The therapy table contains the issued acute prescriptions. Similar to the repeat table, the drug name, dosage, term, form, and event date were extracted.

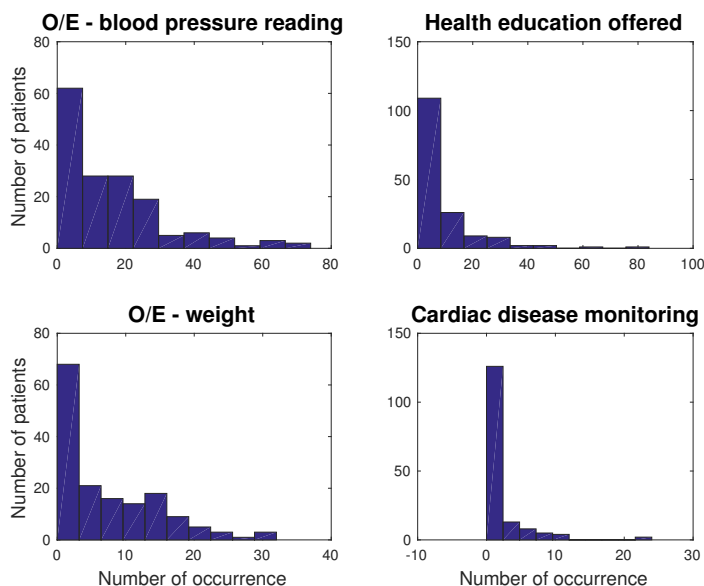


Figure 4.9: Distribution of number of instances of blood pressure, Health education offered, weight and cardiac disease monitoring recorded against patient records

## 4.8.2 Reference validation dataset

We use a standard EHR reference data set for evaluation because it is made up of similar arbitrary-length sequences of symbols. The publicly available UCI machine learning data on membranolytic anticancer peptides (ACPs) (available at [ <https://archive.ics.uci.edu/ml/datasets/Anticancer+peptides>]) was applied as the validation dataset for the published work [184]. The symbolic anticancer peptide data is made up of one-letter amino acid sequences for breast cancer and lung cancer cells. It was previously used in an ensemble machine learning study [108] that identified anticancer peptides. The dataset consists of 4 classes (inactive-exp, inactive-virtual, moderately active, and very active) distributed according to 83, 750, 98, and 18 examples, respectively. The length of sequences ranges between 5 (the minimum) and 38 (the maximum), with a mean length of 17 and a 5.5 standard deviation. No data cleansing or pre-processing steps were performed on the dataset.

The validation aimed at classifying the data into the “inactive-virtual” class by treating the task as a multi-class learning problem. An equal distribution of both classes is applied in order to avoid introducing bias from class imbalance. As a result, the negative class (199) contains all members of the “inactive - exp” (83), “mod active” (98), and “very active” (18) classes, whereas the positive class (199) contains an equal number of examples (199) from the “inactive - virtual” class. An initial partition of 60% for training, 20% for validation, and 20% for testing datasets was adopted for cross validation; however, to ensure a consistent approach was adopted for all experiments, the test and training datasets were combined and the leave-one-out (LOOCV) cross validation adopted instead.

Ten objects were set aside as the set of zero vector sequences and used in constructing kernels via the distance substitution method. Finally, the data classes were distributed according to 156 positive and 152 negative examples.

## 4.9 Summary

In this chapter, we state the research design and methodology. We described the EHR data that is made up of an uneven-length, longitudinal, heterogeneous mixture of symbolic data and numeric test values. We displayed the sample test measurements, showing the mean (std) values that were used in the predictive modelling of a heterogeneous mixture with symbolic EHR data. We also showed an illustration of the distribution of variable-length sequences per data table. The independent, dependent, and multivariate analyses are stated. The edit distance and variants are described in detail, with the justification for their development noted where necessary. Methods of constructing valid kernel functions from the edit distance, the kernel learning algorithms, the multi-kernel learning method, the kernel selection, and evaluation techniques are also described.

# Chapter 5

## Experiments/Results

### 5.1 Description

In this section, we present our findings. We investigate the suitability of the kernel framework with a series of experiments designed to evaluate its predictive capability as a risk prognosis methodology applied to routinely collected EHR. As described in the problem statement, all experimental objectives of this research are based on predicting the likelihood of developing type 2 diabetes given an earlier episode of elevated blood pressure of 130/80 mmHg. The working hypothesis is that elastic edit distance in general models temporality and addresses the problem of irregularly-sampled, uneven-length EHR data, while the bespoke variants described in Section 4.2.1 attempt to enhance the predictive capabilities. We present the experimental objectives and results achieved in this section. The kernel functions were tested using uneven length symbolic sequences extracted from the six data tables (data modalities). Firstly, we experiment on data extracted from the distinct relational data tables they were originally stored in and, secondly, as a collective single view of data for each patient.

The classification performance for all experiments is presented in terms of the F1-score, accuracy, sensitivity, specificity, and number of support vectors. The generated kernels were also examined to determine if they were PSD, thus inducing a RKHS. The kernel target alignment score (KTA), spectral ratio, and average percentage number of negative eigenvalues for the indefinite kernels were also computed. We applied the kernel spectra transformations - flipping, shifting, clipping, and squaring the negative eigenvalues - in addition to classification with the indefinite kernels and noted the best performance achieved. A greedy search approach was adopted to obtain the most suitable zero vector and template candidates for the distance substitution and template matching kernels, respectively, by running the classification process 158 times since all sequences were deemed potential candidates. For each iteration, the candidate example was excluded from the dataset and applied as the zero vector or template sequence. Using Leave-One-Out cross validation, we applied the LibSVM algorithm in a heuristic manner with 4 regularization parameters ( $C$ ) to classify the data. Kernel discriminative power was further improved by using kernel post processing to generate two additional variants of each kernel matrix. First, the classifier was executed on the kernel in its original raw form; second, the kernel was normalised; and lastly, the normalised kernel was centralized. By centralizing the kernels, this work assessed in isolation the usefulness of translation by the zero vector to the origin of the feature space.

### 5.2 Experimental Objectives

- ***Effect of different kernel functions:*** The suitability of bespoke kernel functions derived from an edit distance measure between a pair of EHR sequences was investigated. The edit



distance variants described in the methodology section 4.2.1 are applied to the data, with the discriminative performance achieved via the methods applied in constructing kernels and inducing a RKHS examined. The purpose of this experiment was to establish the right combination that achieved the best results.

- ***Edit distance on sequences with common items:*** This was designed to explore the option of excluding spurious data that may constitute noise and degrade classification performance. Uncommon symbols from a given pair of sequences were first excluded, so they did not contribute to the proximity computation. Items such as gender-specific events may have a negative impact on the similarity between male and female type 2 diabetes patients with identical outcomes. Uncommon symbols in this context would inflate the distance between the pair.
- ***Single vs multiple kernels:*** A single kernel matrix expresses the data distribution and structure within its corresponding induced feature space. Each kernel function, therefore, denotes a different expression of, or window on, the underlying sequential pattern structure. A weighted combination of multiple kernels with MKL allows us to forgo the problem of determining which is the most discriminative of the edit-distance kernel variants. Combining kernels in this manner can be extended to integrate data derived from heterogeneous longitudinal data sources. Nevertheless, this experiment investigates and compares the classification performance of stand-alone single kernels against multiple kernels.
- ***Edit distance on variable-length sequences:*** This experimental objective was designed to determine any impact arbitrary length sequences may have on the similarity of objects. An attempt was made to establish if patterns were easier to extract when the comparative lengths of a given pair of sequences did not differ greatly. Table 4.4 and the illustration in Fig 4.8 show the range of variable lengths across the entire datasets. Filters were applied to limit the gap between the lengths of the sequences. In doing so, examples that did not meet the specified criteria were dropped from the data set. Consequently, four subsets were created. Unfortunately, this further reduced the number of examples available to test with. The classification performance on the full complement of the data (158 examples) was compared to the performance achieved on the additional smaller subsets of sizes 134, 115, and 104 examples.
- ***Multiple kernel learning of heterogeneous entities:*** This experiment explored the possibility of using the kernel framework to address the problem of modelling with heterogeneous data. It combined kernels derived from symbolic data with kernels from real-valued test entities into a single classification model. The purpose of this experiment was to assess the suitability of the kernel framework in the stated context, in terms of the classification performance achieved.
- ***Measure predictive performance of data tables:*** The datasets extracted from distinct relational tables that made up the EHR system offered the opportunity to establish specific tables with the most informative structure. Medical events with diverse attributes are filtered and stored in these data tables for easier processing and management. Certain data tables are likely to contain richer information that can aid in the linear separability of the data, while others may inadvertently hold spurious contents with no predictive value. The available datasets offered this research a variety to experiment with.
- ***Static evaluation of the kernel functions:*** Static measures that can determine the quality of kernels were applied to establish their overall suitability in conjunction with the classification performance attained. Prior knowledge of the discriminatory capabilities of the induced feature space can help in model selection. Consequently, the percentage number of negative eigenvalues, the kernel target alignment, diagonal dominance, and spectral ratio values of each kernel were assessed. In addition, these were also compared to the optimal weighting coefficient achieved with MKL, where applicable.

- **Comparison with traditional bag-of-words (BoW):** To demonstrate the effectiveness of the proposed method, the kernel framework was compared against conventional models based on bag-of-words features derived from the dataset. This served as the baseline to evaluate the predictive performance of the kernel framework as a suitable disease prognosis tool. This objective provided the basis for validation of the published work [184].
- **Validation on external data:** We further validate the suitability of the proposed model via experiments designed to test the robustness of our model as a solution to predictive modelling of uneven-length sequences of symbolic data. This objective provided the basis for validation of the published work [184].

## 5.3 Results / Findings

### 5.3.1 Effect of different kernel functions

In order to verify the suitability of the kernel framework, we evaluated the discriminative performance of the edit distances described in Section 4.2.1 and implemented them with the kernel construction methods defined in Section 4.3.1. To recap, Table 4.3 displays the variants of the edit distances with the kernel construction methods applied. We used the elastic edit distances to address the problem of arbitrary length and irregularly sampled EHR data, while the kernel construction methods presented a means to induce a RKHS. The distances were initially implemented as pseudo kernels, i.e. used directly as the design matrix for the classifier before conversion to valid kernels. The  $\alpha$  parameters for the Gaussian and rational quadratic edit kernels were (0.0005, 0.003, 0.035, 1, 12.5) and (0.025, 1, 1.25), respectively. In order to construct the edit inhomogeneous polynomial kernel, the converted edit similarity rather than the distance measure was applied to replace the traditional dot product of the vector space polynomial kernel. As a result, the adapted edit polynomial kernel was created with degrees (2, 3, and 5) and  $c$  set to 1. We employed a greedy search to find the most suitable zero-vector and template-matching candidate. As a result, the classification process was executed 158 times for both distance substitution and template matching kernel methods, since all examples are potential candidates.

Data	Ker	F1	Acc (%) $\pm$ (std)	Sen	Spec	Mod	Trans	C	nSV	%-ve Eig
Clinical	$k_{ed4}$	0.61	56.33 (49.76)	0.80	0.39	nsd	None	128	58 (1)	49.37
Recall	$k_{ed1}$	0.54	43.04 (49.67)	0.80	0.16	nsd	None	0.25	2 (0)	77.22
Refer	$k_{ed4}$	0.63	61.39 (48.84)	0.80	0.48	square	None	0.0005	123 (2)	63.29
<b>Repeat</b>	<b><math>k_{ed2}</math></b>	<b>0.67</b>	<b>65.82 (47.58)</b>	<b>0.82</b>	<b>0.54</b>	<b>square</b>	<b>None</b>	<b>0.25</b>	<b>56 (1)</b>	<b>93.67</b>
Test	$k_{ed1}$	0.67	63.29 (48.35)	0.88	0.46	clip	None	0.0005	36 (0)	81.01
Therapy	$k_{ed2}$	0.60	55.70 (49.83)	0.79	0.39	square	None	0.25	96 (2)	93.04
All data	$k_{ed1}$	0.60	55.70 (49.83)	0.79	0.39	flip	None	0.0005	32 (0)	99.37

Table 5.1: Best classification performance obtained with edit distance measures implemented as a pseudo kernels, where **Ker** : Kernel functions; **F1** : F1-score; **Acc** : Accuracy; **Sen** : Sensitivity; **Spec** : Specificity; **Mod** : Spectrum Modification; **Trans** : post-kernel Transformation; **C** : SVM C parameter; **nSV** : Number of Support Vectors; **%-ve Eig** : Percentage -Number of negative Eigenvalues

We present the best classification performance obtained with each kernel construction method. Accordingly, in Table 5.1, we display the results obtained with pseudo kernels per dataset. The best F1-score of **0.67** was achieved via the native “edit distance” as a pseudo kernel ( $k_{ed1}$ ) applied to the Test dataset and via “edit distance normalised by the length of the longer sequence” as a pseudo kernel ( $k_{ed2}$ ) on the Repeat dataset. The pseudo-kernel ( $k_{ed2}$ ) executed on the Repeat dataset, on the other

hand, achieved a higher accuracy of **65.82% ± 47.58**. As expected, all pseudo-kernels derived from all datasets were indefinite. The best result was achieved on a kernel without post-matrix normalization. Nevertheless, the kernel had 93.67% of negative eigenvalues; therefore, its spectrum was transformed with the “square” spectrum modification method. The full results achieved with all edit distances implemented as pseudo kernels and applied to the datasets are displayed in Table A.1 of Appendix A.

Data	Ker	F1	Acc (%) ± (std)	Sen	Spec	Mod	Trans	C	nSV	%-ve Eig
Clinical	$k_{s\_ed3}$	0.62	58.86 (49.37)	0.83	0.41	nsd	None	0.25	35 (1)	1.90
Recall	$k_{s\_ed2}$	0.61	48.73 (50.14)	0.94	0.16	nsd	None	0.25	45 (1)	10.13
<b>Refer</b>	<b><math>k_{s\_ed2}</math></b>	<b>0.64</b>	<b>60.76 (48.98)</b>	<b>0.82</b>	<b>0.46</b>	<b>flip</b>	<b>Norm</b>	<b>0.25</b>	<b>75 (1)</b>	<b>4.43</b>
Repeat	$k_{s\_ed4}$	0.62	59.49 (49.25)	0.79	0.46	flip	Norm	128	106 (1)	3.16
Test	$k_{s\_ed4}$	0.60	55.70 (49.83)	0.80	0.38	nsd	None	0.25	32 (0)	50.00
Therapy	$k_{s\_ed3}$	0.51	62.03 (48.69)	0.48	0.72	flip	None	0.25	109 (1)	3.16
All data	$k_{s\_ed9}$	0.61	60.13 (49.12)	0.73	0.51	clip	Norm	128	58 (1)	39.87

Table 5.2: Best classification results obtained with edit similarity kernels

Data	Ker	F1	Acc (%) ± (std)	Sen	Spec	Mod	Trans	C	nSV	%-ve Eig
Clinical	$k_{G\_ed7}$	0.63	58.86 (49.37)	0.86	0.39	clip	Norm	128	71 (1)	5.06
Recall	$k_{G\_ed2}$	0.62	51.27 (50.14)	0.95	0.20	clip	Norm	0.25	38 (1)	12.03
<b>Refer</b>	<b><math>k_{G\_ed2}</math></b>	<b>0.64</b>	<b>62.66 (48.52)</b>	<b>0.80</b>	<b>0.50</b>	<b>None</b>	<b>None</b>	<b>128</b>	<b>17 (0)</b>	<b>7.59</b>
Repeat	$k_{G\_ed2}$	0.66	65.82 (47.58)	0.77	0.58	None	None	128	23 (1)	5.06
Test	$k_{G\_ed4}$	0.63	56.33 (49.76)	0.89	0.33	square	Norm	0.25	9 (0)	50.00
Therapy	$k_{G\_ed2}$	0.62	60.76 (48.98)	0.76	0.50	clip	Norm	128	51 (2)	5.70
All data	$k_{G\_ed4}$	0.64	54.43 (49.96)	0.94	0.26	square	None	0.25	34 (1)	50.00

Table 5.3: Best classification performance achieved with the Gaussian kernel method

Likewise, the best classification performance achieved via the edit similarity kernels is displayed in Table 5.2. The best F1-score of **0.64** and accuracy of **60.76% ± 48.98** were achieved via the “edit distance normalised by the length of the longer sequence” ( $k_{s\_ed2}$ ) and applied to the Refer dataset. The kernel’s spectrum decomposition showed a low 4.43% number of negative eigenvalues and was transformed with the flip spectrum modification method. In addition, this was achieved with the 0.25 SVM C parameter and resulted in 75 support vectors. The entire results achieved with the similarity kernels applied to all datasets are displayed in Table A.2 of Appendix A. In table 5.3, we display the best classification results achieved with the Gaussian edit kernels. The optimum F1-score of **0.64** was achieved via ‘edit distance normalised by the length of the longer sequence’ ( $k_{G\_ed2}$ ) applied to the Refer dataset and “edit distance normalised by the exponent of the number of common items” ( $k_{G\_ed4}$ ) applied to the single view dataset (All data). The kernel function ( $k_{G\_ed2}$ ) applied to the Refer dataset, on the other hand, had a higher accuracy of **62.66% ± 48.52**. The result was achieved with an indefinite kernel with 7.59% negative eigenvalues and an SVM C parameter of 128 that required 17 support vectors. The entire results achieved with the Gaussian kernel construction method are displayed in Table A.3 of Appendix A.

A similar result was achieved with the rational quadratic kernels via the ‘edit distance normalised by the length of the longer sequence’ ( $k_{r\_ed2}$ ) and applied to the Refer table. Table 5.4 shows this with the best F1-score of **0.64** and accuracy of **60.76% ± 48.98**. The kernel had 4.43% number of

Data	Ker	F1	Acc (%) $\pm$ (std)	Sen	Spec	Mod	Trans	C	nSV	%-ve Eig
Clinical	$k_{r\_ed3}$	0.62	58.86 (49.37)	0.83	0.41	nsd	None	0.25	35 (1)	1.90
Recall	$k_{r\_ed2}$	0.62	51.27 (50.14)	0.95	0.20	nsd	None	0.25	42 (1)	10.76
<b>Refer</b>	<b><math>k_{r\_ed2}</math></b>	<b>0.64</b>	<b>60.76 (48.98)</b>	<b>0.82</b>	<b>0.46</b>	<b>flip</b>	<b>Norm</b>	<b>0.25</b>	<b>75 (1)</b>	<b>4.43</b>
Repeat	$k_{r\_ed4}$	0.62	59.49 (49.25)	0.79	0.46	flip	Norm	128	106 (1)	2.53
Test	$k_{r\_ed4}$	0.60	55.70 (49.83)	0.80	0.38	nsd	None	0.25	32 (0)	50.00
Therapy	$k_{r\_ed10}$	0.57	48.10 (50.12)	0.82	0.24	psd	None	128	18 (0)	0.00
All data	$k_{r\_ed9}$	0.63	60.76 (48.98)	0.79	0.48	clip	None	128	58 (1)	39.24

Table 5.4: Best classification performance achieved with the rational quadratic kernel method

negative eigenvalues, and its spectrum was transformed with the flip spectrum modification method. This was achieved with an SVM C parameter of 0.25 that resulted in 75 support vectors. The entire results obtained with the rational quadratic kernels are displayed in Table A.4 of Appendix A. The best F1-score of **0.64** and accuracy of **61.39%  $\pm$  48.84** achieved with the polynomial edit kernel implemented via “edit distance normalised by the length of the longer sequence” ( $k_{p\_ed2}$ ) and applied to the Refer table are shown in Table 5.5. The post-normalized indefinite kernel with 5.70% number of negative eigenvalues was transformed with the flip spectrum modification method. The result was also achieved with an SVM C parameter of 0.25 and 64 support vectors. See Table A.5 of Appendix A for the entire results obtained with the edit polynomial kernels.

Data	Ker	F1	Acc (%) $\pm$ (std)	Sen	Spec	Mod	Trans	C	nSV	%-ve Eig
Clinical	$k_{p\_ed3}$	0.63	60.76 (48.98)	0.79	0.48	nsd	None	0.25	42 (0)	1.90
Recall	$k_{p\_ed2}$	0.61	48.73 (50.14)	0.94	0.16	flip	None	0.25	50 (1)	10.13
<b>Refer</b>	<b><math>k_{p\_ed2}</math></b>	<b>0.64</b>	<b>61.39 (48.84)</b>	<b>0.82</b>	<b>0.47</b>	<b>clip</b>	<b>Norm</b>	<b>0.25</b>	<b>64 (1)</b>	<b>5.70</b>
Repeat	$k_{p\_ed2}$	0.63	66.46 (47.36)	0.70	0.64	flip	None	0.25	101 (1)	2.53
Test	$k_{p\_ed4}$	0.60	58.86 (49.37)	0.74	0.48	clip	Norm	128	36 (1)	46.84
Therapy	$k_{p\_ed3}$	0.53	65.19 (47.79)	0.47	0.78	flip	None	0.25	111 (0)	1.90
All data	$k_{p\_ed9}$	0.63	61.39 (48.84)	0.79	0.49	clip	None	0.25	58 (1)	39.24

Table 5.5: Best classification performance achieved with the polynomial kernel method

Data	Ker	$X_0$	F1	Acc (%) $\pm$ (std)	Sen	Spec	Mod	Trans	C	nSV	%-ve Eig
<b>All data</b>	<b><math>k_{t\_ed1}</math></b>	<b>127</b>	<b>0.60</b>	<b>49.04 (50.15)</b>	<b>0.91</b>	<b>0.19</b>	<b>nsd</b>	<b>Norm</b>	<b>0.25</b>	<b>4 (0)</b>	<b>49.04</b>

Table 5.6: Best classification performance achieved with the template matching method. Where  $X_0$  : Template candidate

In Table 5.7, we display the best performance achieved with the distance substitution kernel method. The best F1-score of **0.74** was achieved on the clinical and full complement datasets (all data) via the ‘edit distance normalised by the length of the longer sequence’ ( $k_{ds\_ed4}$ ). However, a higher accuracy of **75.80%  $\pm$  42.97** was achieved on the clinical dataset. The post-normalized indefinite matrix had 22.29% of negative eigenvalues and was transformed with the flip-spectrum modification method. An SVM C parameter of 0.25 resulted in 40 support vectors. Furthermore, we display the entire results achieved with the distance substitution method in Appendix A.6. Likewise, the best result achieved with the template matching method applied to the single view data set (all data) is displayed in 5.6. With example 127 as the template candidate, an F1-score of **0.60** and an accuracy of **49.04%  $\pm$  50.15** were obtained. An SVM parameter of 0.25 resulted in 4 support vectors. The normalised indefinite kernel had 49.04% of negative eigenvalues.

Data	Ker	X <sub>0</sub>	F1	Acc (%) ± (std)	Sen	Spec	Mod	Trans	C	nSV	%-ve Eig
<b>Clinical</b>	<b>k<sub>ds_ed4</sub></b>	<b>121</b>	<b>0.74</b>	<b>75.80 (42.97)</b>	<b>0.80</b>	<b>0.73</b>	<b>flip</b>	<b>Norm</b>	<b>0.25</b>	<b>40 (1)</b>	<b>22.29</b>
Recall	<i>k<sub>ds_ed4</sub></i>	13	0.63	52.23 (50.11)	0.95	0.21	flip	Norm	0.25	34 (1)	24.84
Refer	<i>k<sub>ds_ed2</sub></i>	8	0.65	61.15 (48.90)	0.85	0.45	clip	None	128	52 (1)	2.55
Repeat	<i>k<sub>ds_ed3</sub></i>	90	0.69	69.43 (46.22)	0.82	0.60	clip	None	0.0005	90 (1)	3.82
Test	<i>k<sub>ds_ed4</sub></i>	102	0.68	63.06 (48.42)	0.95	0.40	clip	None	128	102 (0)	49.04
Therapy	<i>k<sub>ds_ed1</sub></i>	55	0.67	66.88 (47.22)	0.80	0.58	psd	norm	0.25	89 (1)	0.00
All data	<i>k<sub>ds_ed4</sub></i>	18	0.74	70.70 (45.66)	0.98	0.51	square	Norm	0.0005	112 (1)	21.02

Table 5.7: Best classification performance achieved with the distance substitution method. Where **X<sub>0</sub>** : Zero Vector Example

Data	Ker	F1	Acc (%) ± (std)	Sen	Spec	Mod	Trans	C	nSV	%-ve Eig
Pseudo	<b>k<sub>ed2</sub></b>	0.67	65.82 (47.58)	0.82	0.54	square	None	0.25	56 (1)	93.67
Similarity	<b>k<sub>s_ed2</sub></b>	0.64	60.76 (48.98)	0.82	0.46	flip	Norm	0.25	75 (1)	4.43
Gaussian	<b>k<sub>G_ed2</sub></b>	0.66	65.82 (47.58)	0.77	0.58	nsd	None	128	23 (1)	5.06
Rat Quad	<b>k<sub>r_ed2</sub></b>	0.64	60.76 (48.98)	0.82	0.46	flip	Norm	0.25	75 (1)	4.43
Poly	<b>k<sub>p_ed2</sub></b>	0.64	61.39 (48.84)	0.82	0.47	clip	Norm	0.25	64 (1)	5.70
<b>Dist Sub</b>	<b>k<sub>ds_ed4</sub></b>	<b>0.74</b>	<b>75.80 (42.97)</b>	<b>0.80</b>	<b>0.73</b>	<b>flip</b>	<b>Norm</b>	<b>0.25</b>	<b>40 (1)</b>	<b>22.29</b>
Temp Mat	<b>k<sub>t_ed4</sub></b>	0.60	49.04 (50.15)	0.91	0.19	nsd	Norm	0.25	4 (0)	49.04

Table 5.8: Best classification performance displayed according to the kernel construction method

Method	Ker	F1	Acc (%) ± (std)	Sen	Spec	Mod	Trans	C	nSV	%-ve Eig
Pseudo	<i>k<sub>ed1</sub></i>	0.62	51.90 (50.12)	0.95	0.21	clip	Norm	128	23 (1)	69.62
Similarity	<i>k<sub>s_ed4</sub></i>	0.60	51.90 (50.12)	0.86	0.27	clip	Norm	0.25	26 (1)	50.63
Gaussian	<i>k<sub>G_ed4</sub></i>	0.62	53.80 (50.01)	0.89	0.28	square	None	0.25	17 (0)	50.00
<b>Rat Quad</b>	<b>k<sub>r_ed4</sub></b>	<b>0.63</b>	<b>55.06 (49.90)</b>	<b>0.91</b>	<b>0.29</b>	<b>nsd</b>	<b>None</b>	<b>0.25</b>	<b>39 (0)</b>	<b>50.63</b>
Poly	<i>k<sub>p_ed4</sub></i>	0.62	54.43 (49.96)	0.86	0.32	nsd	None	0.0005	107 (1)	50.00

Table 5.9: Best classification performance achieved with the kernel construction methods applied to common symbols

Finally, a comparison of the best results (F1-score and accuracy ) achieved per kernel construction method is illustrated via bar plots in 5.1 and Table 5.8. The distance substitution method outperformed the other methods in the bar plot, with an F1-score of **0.74** and an accuracy of **75.80% ± 42.97**. Next, we present the performance achieved with the kernel methods applied via the native ‘edit distance’, ‘edit distance normalised by the length of the longer sequence’, ‘edit distance normalised by the number of common items’, ‘edit distance normalised by the exponent of the number of common items’, and ‘edit distance computed with number of unmatched vs matched symbols’ applied to the same single view dataset (all data). The plot 5.3 shows the distance substitution method clearly outperformed other kernel methods on the same dataset.

### 5.3.2 Edit distance on sequences with common items

Given a pair of sequences, symbols that only exist in one of the sequences were discarded prior to computing the edit distance between the common symbols remaining in both records. It is important to note that the sequential order of the symbols in both sequences was retained in the process. This experiment provided a means to test the effect of spurious information that was unlikely to contribute to the predictive performance and was performed with the full complement of the data (all data). The bar chart in 5.4 shows the predictive performance in terms of F1-score and accuracy achieved via

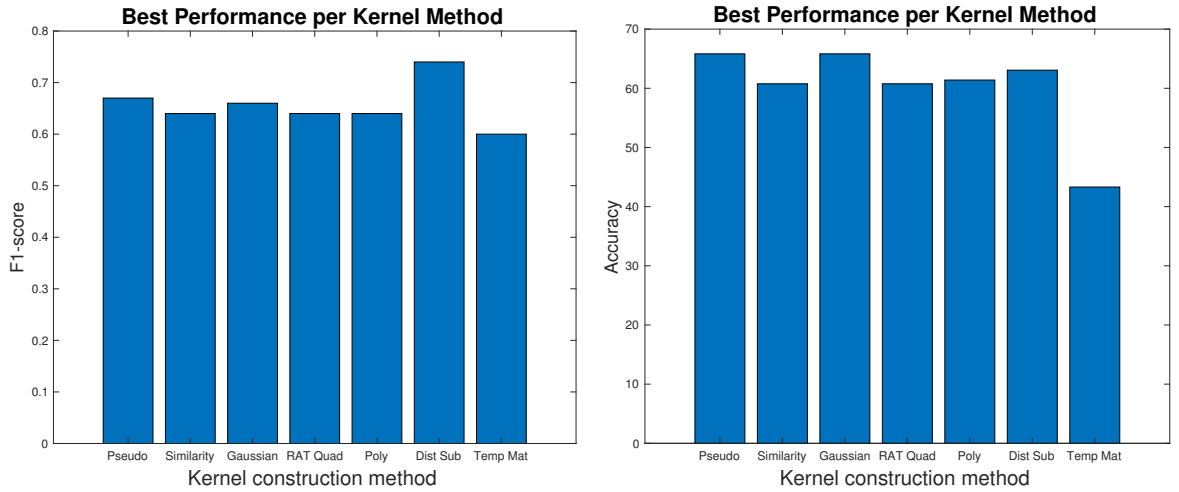


Figure 5.1: Comparison of the best performance achieved per kernel construction method and applied to Clinical, Recall, Refer, Repeat, Test, Therapy and All data datasets.

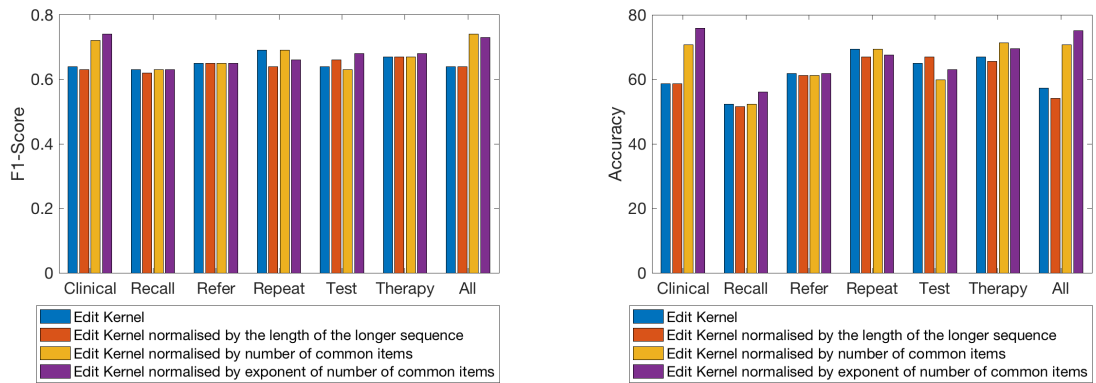


Figure 5.2: F1-Score and Accuracy achieved with distance substitution kernel construction method applied to the datasets

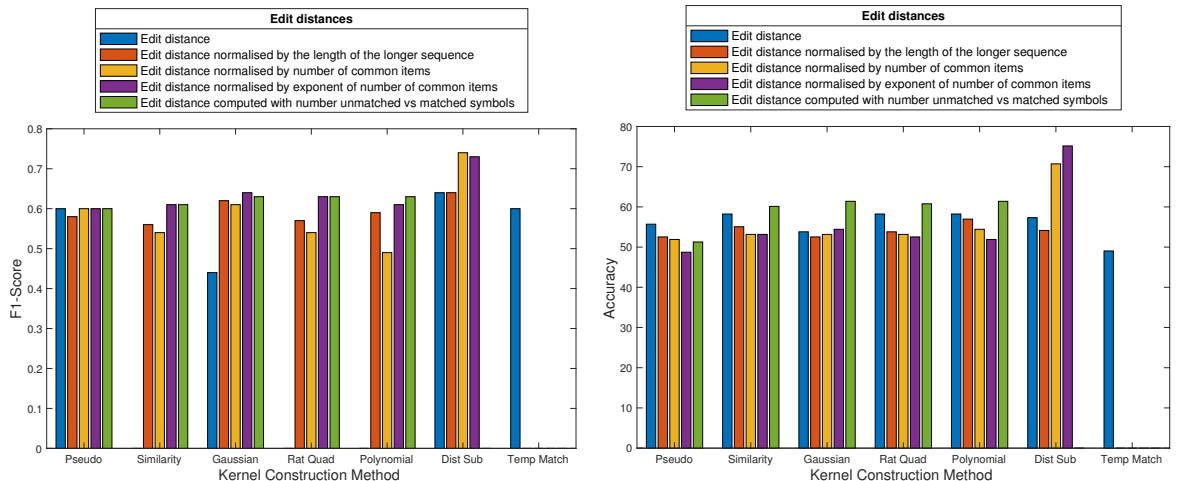


Figure 5.3: F1-score and accuracy achieved with 7 kernel construction methods via 5 edit distance variants applied to the single view dataset (Alldata)

4 edit distances implemented with the kernel construction methods. The rational quadratic kernel implementing the "edit distance normalised by the exponent of the number of common items" ( $\mathbf{k}_{r\_ed4}$ ) achieved the best F1-score of **0.63** and accuracy of **55.06% ± 49.90**. The SVM C parameter of 0.25

resulted in 39 support vectors on an indefinite kernel matrix with 50.63% negative eigenvalues. The entire raw results obtained are displayed in Table A.14 in Appendix A.

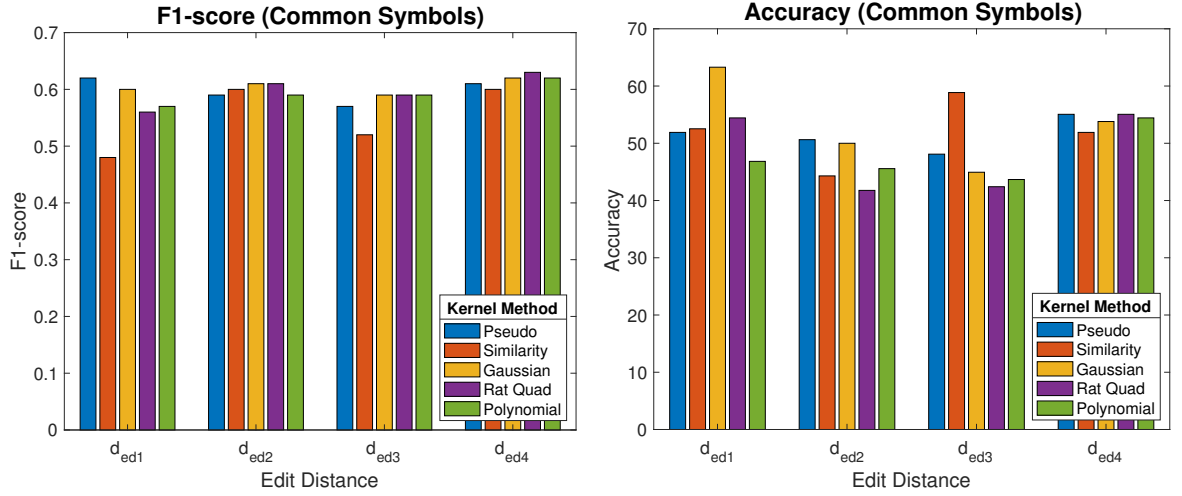


Figure 5.4: F1-Score and Accuracy achieved by applying the kernels on symbols common to the pair of sequences via native ‘edit distance’ ( $d_{ed1}$ ), ‘edit distance normalised by the length of the longer sequence’ ( $d_{ed2}$ ), ‘edit distance normalised by the the number of common items’ ( $d_{ed3}$ ), and ‘edit distance normalised by the exponent of number of common items’ ( $d_{ed4}$ )

### 5.3.3 Single vs multiple kernels:

Combining kernels algebraically into a single model offers the possibility of an enhanced representation of the patterns we are seeking to exploit. Using SimpleMKL, we sought the optimum linear combination of multiple kernels. In some experiments, the multiple kernels were generated from the same kernel method via the same edit distance by varying the kernel hyper-parameters, such as with the rational quadratic, Gaussian, and polynomial edit kernels. These methods generate multiple variants by varying their parameters. In other MKL experiments, a mixture of kernels derived from different methods and edit distances was combined. However, the distance substitution and template matching kernels were created by using one of the data points as the zero vector and template matching candidate; thus, with one less data point, they could not be combined with kernels derived from the other methods. In addition, MLK allowed the combination of kernels derived from distinct datasets into a single model. It also provided the means to address modelling with a heterogeneous combination of symbolic and numeric EHR data.

Tables	$X_0$	F1	Acc(%)	Sen	Spec	nSV
Clinical	88	0.78	78.34	0.89	0.70	112
Recall	44	0.95	96.18	0.91	1.00	124
Refer	97	0.95	96.18	0.91	1.00	127
Repeat	85	0.95	96.18	0.91	1.00	127
Test	110	0.73	70.06	0.97	0.51	75
Therapy	3	0.72	68.79	0.97	0.49	105
<b>All data</b>	<b>37</b>	<b>0.96</b>	<b>96.82</b>	<b>0.95</b>	<b>0.98</b>	<b>116</b>

Table 5.10: Best results obtained from MKL convex optimization combining the four kernels ( $\mathbf{k}_{ds\_ed1}$ ,  $\mathbf{k}_{ds\_ed2}$ ,  $\mathbf{k}_{ds\_ed3}$ , and  $\mathbf{k}_{ds\_ed4}$ ) applied independently to the respective datasets

Firstly, we compared and evaluated the performance achieved via single kernel learning vs. MKL using the distance substitution kernel construction method. This specific set of experiments served as the foundation for the published work [184]. Table 5.10 displays the MKL results obtained for each

dataset. The single view dataset produced an F1-score of **0.96**, whereas the recall, refer, and repeat datasets produced an F1-score of **0.95**. The plots in Figure 5.5 compare the F1-score and accuracy of single kernel performance vs. MKL performance. We also show the learned MKL combination weight coefficients  $\sigma$  in Table 5.12. The entire 24 kernel matrices resulting from 4 distance substitution edit kernels applied to the clinical, Recall, Refer, Repeat, Test, and Therapy datasets were also combined via MKL into a single classification model. This model achieved an F1-score of **0.92** as can be seen in Table 5.11. The target alignment scores achieved with the kernels are also displayed in Table 5.13.

Tables	$X_0$	F1	Acc	Sen	Spec	nSV
All kernels	90	0.92	94.27	0.86	1	41

Table 5.11: MKL results obtained from combining 24 kernel matrices derived from the datasets. (The 4 kernels -  $\mathbf{k}_{ds\_ed1}$ ,  $\mathbf{k}_{ds\_ed2}$ ,  $\mathbf{k}_{ds\_ed3}$ , and  $\mathbf{k}_{ds\_ed4}$  per dataset)

Tables	$\mathbf{k}_{ds\_ed1}$	$\mathbf{k}_{ds\_ed2}$	$\mathbf{k}_{ds\_ed3}$	$\mathbf{k}_{ds\_ed4}$
Clinical	0.2259	0.2261	0.2261	0.3219
Recall	0.2177	0.2322	0.2390	0.3110
Refer	0.1831	0.1988	0.2030	0.4151
Repeat	0.1562	0.1591	0.1599	0.5248
Test	0.2500	0.2500	0.2500	0.2500
Therapy	0.0000	0.0000	0.0000	1.0000
All data	0.0000	0.0000	0.0000	1.0000

Table 5.12: Weights (sigma) obtained from multi kernel learning process with classification performance displayed in Table 5.10

We also compared MKL classification performance in three additional experiments, the results of which are shown in Figure 5.6. The experiment (**Exp 1**) compared the results achieved via kernels applied to common symbols in the single view dataset. A higher F1-score of **0.89** and accuracy of **91.14%  $\pm$  28.51** was achieved with MKL (see table 5.15) compared to an F1-score of **0.63** and accuracy of **55.06%  $\pm$  49.90** achieved via a single rational quadratic edit kernel, as displayed in Table 5.9. The experiment (**Exp 2**) compares the F1 score of **0.98** and accuracy of **98.73%  $\pm$  11.21** (see Table 5.16) achieved by combining multiple Gaussian edit kernels via the ‘edit distance computed with the number of match and unmatched symbols’ ( $d_{ed9}$ ) applied to the single view dataset. This was compared to the polynomial edit kernel’s best F1-score of **0.63** and accuracy of **61.39%  $\pm$  48.84** on the same dataset (See Appendix A.5). Finally, the experiment (**Exp 3**) compared the MKL results obtained by applying the “edit distance with controlled equality” ( $d_{ed10}$ ) to the Therapy dataset. An F1-score of **0.95** and accuracy of **96.20%  $\pm$  19.17** were achieved via MKL (See Table 5.17). This result outperformed the F1-score of **0.59** and **42.41%  $\pm$  49.58** accuracy achieved with a single kernel via the same distance implemented as a pseudo kernel. (See the Table A.1).

Table	$\mathbf{k}_{ds\_ed1}$	$\mathbf{k}_{ds\_ed2}$	$\mathbf{k}_{ds\_ed3}$	$\mathbf{k}_{ds\_ed4}$
Clinical	0.9735	0.9577	0.8862	0.6356
Recall	0.9212	0.9157	0.9065	0.8917
Refer	0.9833	0.9812	0.9797	0.9579
Repeat	0.9825	0.9728	0.9721	0.9124
Test	0.8383	0.8119	0.7607	0.5562
Therapy	0.9929	0.9926	0.9208	0.5379
All data	0.9577	0.9169	0.8219	0.2378

Table 5.13: Kernel Target Alignment scores obtained with the 4 kernels derived with the distance substitution methods applied to all datasets. This shows the alignment with the target labels for the results displayed in Table 5.10



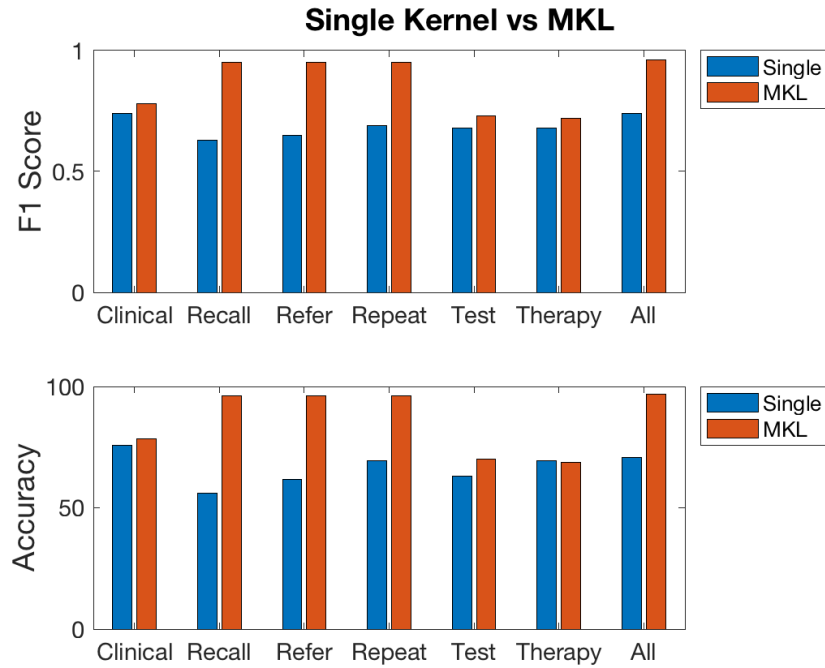


Figure 5.5: Best F1-Score and Accuracy achieved with a single kernel vs MKL using the distance substitution kernel construction method. This compares results from stand-alone single kernels displayed in Table 5.7 against the MKL results in Table 5.10

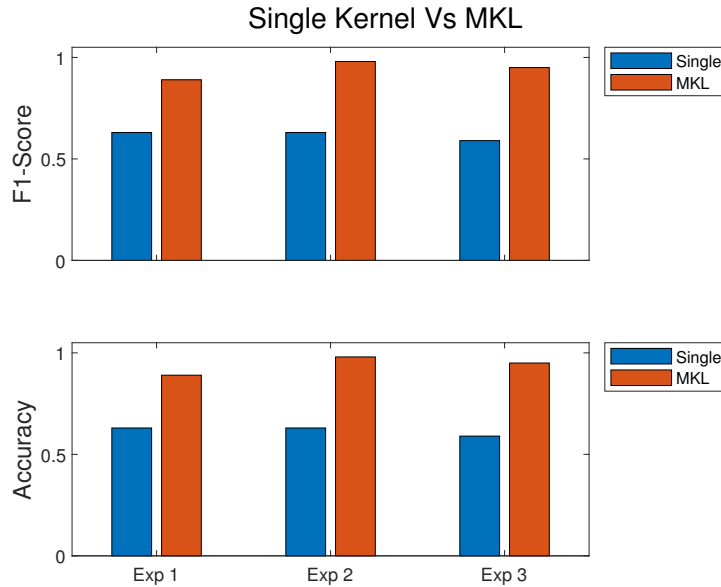


Figure 5.6: Comparison of performance achieved with single kernel vs MKL. Where **Exp 1**: Compares MKL vs single kernels applied to common symbols from the single view dataset - see 5.15 where the MKL results are displayed. **Exp 2**: Refers to experiments with kernel derived from edit distance computed with number of matched and unmatched symbol ( $d_{ed9}$ ) applied to the single view dataset. See 5.16 for the MKL results. **Exp 3**: Experiments via Edit distance with controlled equality ( $d_{ed10}$ ) applied to the therapy dataset. See Table 5.17

### 5.3.4 Edit distance on variable-length sequences:

The sequence length variation experiment was designed to see if arbitrary length sequences affected the similarity of two sequences. Although the edit distance as an elastic measure can deal with distortions, having closer length sequences may reduce the effects of large gaps on the calculated proximity measure. The Table 4.4 displays the aggregate length description (minimum, maximum,

and mean length of sequences per table). The selection criteria for this experiment were based on the Clinical dataset since it contains the journal view of medical events recorded. The first tranche was treated exactly as recorded, i.e., with 158 examples. Next, cohorts of patients with sequence lengths greater than 40, 75, and 90 symbols are selected. Unfortunately, by excluding patients that did not meet the specified criteria, the dataset size was reduced to 134, 115, and 104 examples, respectively.

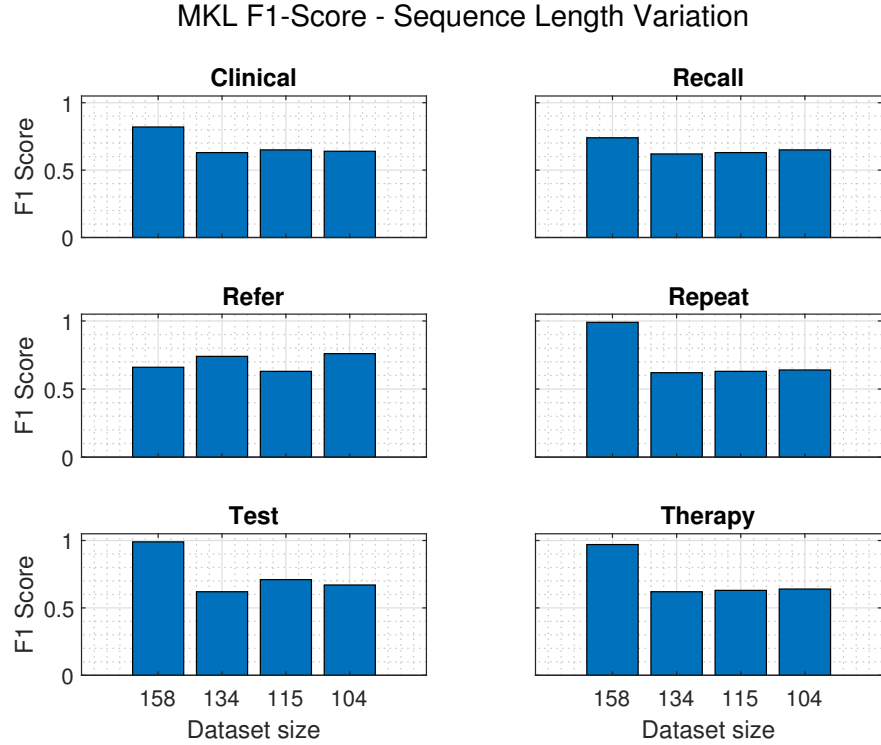


Figure 5.7: MKL F1-Score achieved from sequence length variation experiment

We carried out the experiments via the native ‘edit distance’ ( $d_{ed1}$ ), ‘edit distance with length normalization’ ( $d_{ed2}$ ), ‘edit distance normalized by the number of common items’ ( $d_{ed3}$ ), and ‘edit distance normalized by the exponent of the number of common items’ ( $d_{ed4}$ ). We combined 1 pseudo kernel and 11 Gaussian edit kernels derived per the 4 edit distance measures with  $\gamma$  parameters - (0.00003, 0.0001, 0.003, 0.0012, 0.035, 0.01, 0.5, 0.1, 3, 12.5, 1250) and applied them to the datasets (Clinical, Recall, Refer, Repeat, Test and Therapy). 12 kernels in total per dataset and per edit distance were derived for each dataset size (158, 134, 115, and 104) and combined via MKL. The best results achieved for each dataset are displayed in Table 5.14 (see the Tables A.8, A.9, A.10, A.11, A.12, and A.13 of Appendix A for the full results obtained with the datasets); the comparison of the F1-scores achieved for the dataset sizes (158, 134, 115, and 104) is clearly illustrated with bar plots in Figure 5.7. As can be seen in the plot, the best results were obtained with a dataset size of 158. The Refer dataset was the only exception where the smallest size (104) had the highest F1-score. Similarly, in Table 5.15, we display the results obtained by applying a combination of pseudo kernels and 8 Gaussian edit kernels with parameters  $\gamma$  - (0.00003, 0.0001, 0.0012, 0.003, 0.035, 0.01, 0.5, 0.1) derived via the same 4 edit distances. This is the experiment (**Exp 1**) depicted in Figure 5.6. Figure 5.8 shows the learned MKL weights  $\sigma$ . As can be seen, the dataset sizes 158, 134, and 104 result in a sparse solution with a single kernel with the heaviest coefficient. Conversely, the learned weights for size 115 are evenly distributed.

The results obtained by combining the pseudo and Gaussian edit kernels via the edit distance computed with the number of unmatched and matched symbols ( $d_{ed9}$ ) with the following 5 - (0.025, 1.01, 1.25, 2, 2.25)  $\tau$  values are shown in Table 5.16. For each  $\tau$  value and pseudo kernel, 8 additional

Dataset	Size	Kernel	F1	Acc(%) $\pm$ (std)	Sen	Spec	C	nSV
Clinical	<b>158</b>	<b><math>k_{G\_ed4}</math></b>	<b>0.82</b>	<b>83.54 (37.20)</b>	<b>0.91</b>	<b>0.78</b>	<b>2.00E+00</b>	<b>2 (1)</b>
	134	$k_{G\_ed4}$	0.63	47.76 (50.14)	0.98	0.07	2.00E+00	3 (2)
	115	$k_{G\_ed1}$	0.65	50.43 (50.22)	1.00	0.08	3.25E-05	53 (4)
	104	$k_{G\_ed4}$	0.64	48.08 (50.20)	0.98	0.04	2.00E+00	10 (2)
Recall	<b>158</b>	<b><math>k_{G\_ed1}</math></b>	<b>0.74</b>	<b>71.52 (45.28)</b>	<b>0.97</b>	<b>0.53</b>	<b>7.05E-03</b>	<b>136 (6)</b>
	134	$k_{G\_ed1}$	0.62	45.52 (49.99)	1.00	0.01	2.00E+00	3 (0)
	115	$k_{G\_ed1}$	0.63	46.96 (50.13)	1.00	0.02	2.00E+00	3 (0)
	104	$k_{G\_ed1}$	0.65	48.08 (50.20)	1.00	0.02	2.00E+00	3 (0)
Refer	158	$k_{G\_ed4}$	0.66	55.70 (49.83)	1.00	0.24	7.05E-03	133 (16)
	134	$k_{G\_ed1}$	0.74	69.40 (46.25)	0.97	0.47	7.05E-03	105 (7)
	115	$k_{G\_ed1}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	2 (1)
	<b>104</b>	<b><math>k_{G\_ed3}</math></b>	<b>0.76</b>	<b>70.19 (45.96)</b>	<b>1.00</b>	<b>0.44</b>	<b>7.05E-03</b>	<b>83 (5)</b>
Repeat	<b>158</b>	<b><math>k_{G\_ed1}</math></b>	<b>0.99</b>	<b>99.37 (7.96)</b>	<b>0.98</b>	<b>1.00</b>	<b>7.05E-03</b>	<b>153 (2)</b>
	134	$k_{G\_ed1}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	2 (0)
	115	$k_{G\_ed1}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	1 (0)
	104	$k_{G\_ed1}$	0.64	47.12 (50.16)	1.00	0.00	2.00E+00	1 (0)
Test	<b>158</b>	<b><math>k_{G\_ed1}</math></b>	<b>0.99</b>	<b>99.37 (7.96)</b>	<b>0.98</b>	<b>1.00</b>	<b>7.05E-03</b>	<b>113 (6)</b>
	134	$k_{G\_ed3}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	2 (0)
	115	$k_{G\_ed3}$	0.71	65.22 (47.84)	0.94	0.40	1.05E-04	86 (17)
	104	$k_{G\_ed1}$	0.67	63.46 (48.39)	0.78	0.51	1.05E-04	51 (3)
Therapy	<b>158</b>	<b><math>k_{G\_ed1}</math></b>	<b>0.97</b>	<b>97.47 (15.76)</b>	<b>0.94</b>	<b>1.00</b>	<b>3.25E-05</b>	<b>37 (4)</b>
	134	$k_{G\_ed1}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	2 (0)
	115	$k_{G\_ed1}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	2 (0)
	104	$k_{G\_ed1}$	0.64	47.12 (50.16)	1.00	0.00	2.00E+00	2 (0)

Table 5.14: Best MKL classification performance achieved by varying the length of the sequences.

Kernel	No	Dataset	Size	F1	Acc(%) $\pm$ (std)	Sen	Spec	C	nSv
$k_{ed1}$	36	All data	<b>158</b>	<b>0.89</b>	<b>91.14 (28.51)</b>	<b>0.91</b>	<b>0.91</b>	<b>2.00</b>	<b>2 (0)</b>
$k_{G\_ed1}$		(Common -	134	0.62	44.78 (49.91)	1.00	0.00	2.00	2 (0)
$k_{G\_ed2}$		symb)	115	0.63	45.21 (50.00)	0.98	0.00	2.00	2 (0)
$k_{G\_ed3}$			104	0.64	46.15 (50.09)	0.98	0.00	2.00	2 (0)

Table 5.15: MKL combination of Pseudo and Gaussian Edit kernels applied to common symbols from the single view dataset via the 4 distances ( $d_{ed1}$ ,  $d_{ed2}$ ,  $d_{ed3}$ , and  $d_{ed4}$ ). The classification results displayed according to dataset sizes - 158, 134, 115, and 104 data points. Where **No**: Number of Kernels combined

Kernel	No	Dataset	Size	F1	Acc(%) $\pm$ (std)	Sen	Spec	C	nSv
$k_{ed9}$	45	All data	<b>158</b>	<b>0.98</b>	<b>98.73 (11.21)</b>	<b>0.98</b>	<b>0.99</b>	<b>7.05E-03</b>	<b>1 (0)</b>
$k_{G\_ed9}$			134	0.56	38.61 (48.84)	0.82	0.01	2.00E+00	1 (0)
			115	0.58	41.77 (49.48)	0.83	0.01	7.05E-03	1 (0)
			104	0.62	44.94 (49.90)	0.84	0.01	7.05E-03	1 (0)

Table 5.16: MKL with Gaussian ‘edit distance computed with unmatched and matched symbols’ ( $d_{ed9}$ ) applied to single view dataset (all data). The classification results are displayed according to dataset sizes - 158, 134, 115, and 104 data points

Gaussian kernels were created with  $\gamma$  (0.00003, 0.0001, 0.0012, 0.003, 0.035, 0.01, 0.5, 0.1). A total of 45 kernels were combined, and the results were obtained on the single-view dataset sizes (158, 134, 115, and 104). This is the experiment (**Exp 2**) depicted in Figure 5.6. The learned MKL weight coefficient  $\sigma$  is depicted in Figure 5.9. All four data sizes had similar weight distributions. It is worth

Kernels	No	Dataset	Size	F1	Acc(%) $\pm$ (std)	Sen	Spec	C	nSv
$k_{ed10}$	36	Therapy	<b>158</b>	<b>0.95</b>	<b>96.20 (19.17)</b>	<b>0.98</b>	<b>0.95</b>	<b>7.05E-03</b>	<b>144 (31)</b>
$k_{G\_ed10}$			134	0.56	38.61 (48.84)	0.82	0.01	1.05E-04	14 (7)
			115	0.58	41.14 (49.37)	0.83	0.00	2.00E+00	2 (0)
			104	0.61	44.30 (49.83)	0.84	0.00	2.00E+00	2 (0)

Table 5.17: MKL result obtained on the therapy data. Classification results displayed according to dataset sizes - 158, 134, 115, and 104 data points

noting that the full-length sequences (i.e., size = 158) achieved the highest F1-score of **0.98** and accuracy of **98.73%  $\pm$  11.21**. The full-length sequences achieved a similar high F1-score of **0.95** and accuracy of **96.20%  $\pm$  19.17**, as shown in Table 5.17. This was achieved by combining 4 pseudo and 32 Gaussian ‘edit distance with controlled equality’ ( $d_{ed10}$ ) applied to the therapy dataset. The edit kernels were generated with the  $h$  parameter set to (183, 365, 548, 730) and the Gaussian  $\gamma$  parameters set to (0.00003, 0.0001, 0.0012, 0.003, 0.035, 0.01, 0.5, 0.1). This is the experiment (**Exp 3**) depicted in Figure 5.6. Figure 5.10 shows the corresponding learned MKL weight coefficient  $\sigma$ . This shows a similar distribution of weights that favours a sparse solution except for the dataset size of 158, where the gap between the largest coefficient and the rest is very small.

Kernel	No	Dataset	Size	F1	Acc(%) $\pm$ (std)	Sen	Spec	C	nSv
$k_{G\_ed11}$	3240	Test	<b>158</b>	<b>0.98</b>	<b>98.73 (11.21)</b>	<b>0.98</b>	<b>0.99</b>	<b>2.00E+00</b>	<b>19 (1)</b>
			134	0.67	53.16 (50.06)	1.00	0.13	7.05E-03	48 (13)
			115	0.69	56.33 (49.76)	1.00	0.14	7.05E-03	43 (16)
			104	0.72	59.49 (49.25)	1.00	0.15	7.05E-03	41 (17)

Table 5.18: MKL with a combination of Gaussian Edit distance kernels computed on 30 numeric Test measurements. Classification results displayed according to dataset sizes - 158, 134, 115, and 104 data points

Kernel	No	Dataset	Size	F1	Acc(%) $\pm$ (std)	Sen	Spec	C	nSv
$k_{G\_ed11}$	810	Test	<b>158</b>	<b>1.00</b>	<b>100.00 (0.00)</b>	<b>1.00</b>	<b>1.00</b>	<b>2.00E+00</b>	<b>2 (0)</b>
			134	0.67	53.16 (50.06)	1.00	0.13	7.05E-03	2 (0)
			115	0.69	89.87 (30.26)	0.88	0.91	3.25E-05	70 (42)
			104	0.72	59.49 (49.25)	1.00	0.15	7.05E-03	2 (0)

Table 5.19: MKL with a combination of 810 Gaussian edit distance kernels computed on 30 numeric Test measurements. Classification results displayed according to dataset sizes - 158, 134, 115, and 104 data points

In Table 5.18, we display the MKL results obtained by combining 3240 kernels derived from 30 numeric test measurements extracted from the Test dataset. With the  $\delta$  parameter (0.1, 0.5, 1), three variants of the ‘edit distance on real sequence’ ( $d_{ed11}$ ) were implemented. First, we apply this to the real-valued test measurements; second, we apply the distance measure to the time axis of the recorded test measurements; and lastly, we add the two to create the third proximity measure. Gaussian edit kernels were created with  $\lambda$  parameters of (0.00003, 0.0001, 0.003, 0.0012, 0.035, 0.01, 0.5, 0.1, 3, 10, 100, 1250). With the full-length sequences (i.e., dataset size = 158), an F1 score of **0.98** and an accuracy of **98.73%  $\pm$  11.21** were obtained. The corresponding learned weight coefficient  $\sigma$  in Table 5.20 supports a sparse solution. The first kernel had the highest coefficient. Likewise, we display the MKL results obtained by combining 810 kernels derived in a similar manner from the 30 test measurements using variants of the ‘edit distance on real sequences’ ( $d_{ed11}$ ) and implemented with the  $\delta$  parameter (0.1, 0.5, 1) in Table 5.19. With  $c$  set to one, the computed edit distance was implemented as an inhomogeneous polynomial edit kernel. In a similar fashion, we implement 3 variants by applying this to the real-valued test measurements, then to the time axis of the test measurement, and lastly, by adding the two to create the third proximity measure. The polynomial edit kernels were implemented

with degrees  $d$  (0.025, 0.5, 2.5). As shown in Table 5.21, the corresponding learned MKL weight coefficients,  $\sigma$ , are more evenly distributed.

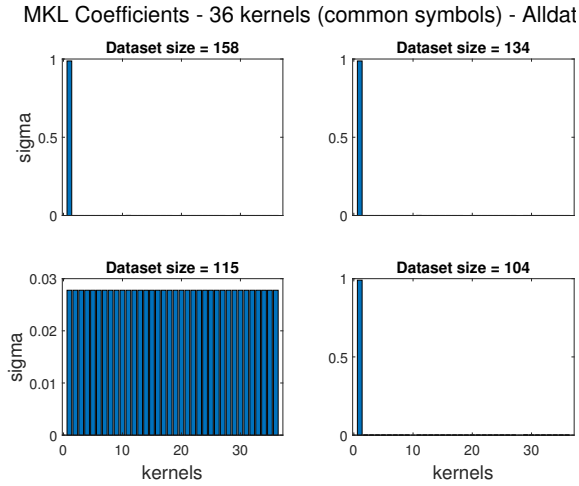


Figure 5.8: MKL coefficients learned from combining 36 Gaussian Edit kernels applied to common symbols and applied to the single view dataset. This corresponds to the MKL results displayed in Table 5.15

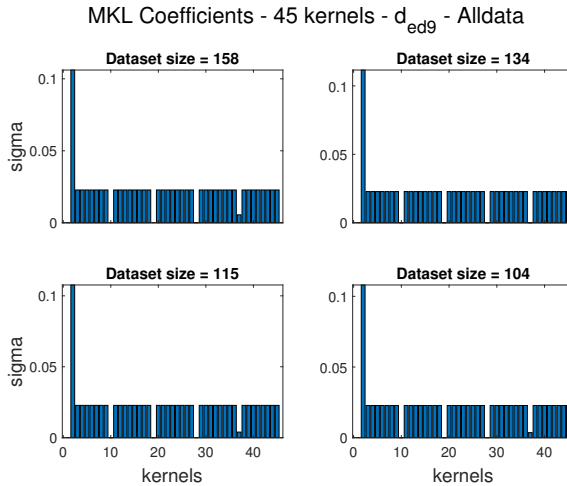


Figure 5.9: MKL coefficients learned from combining 45 Gaussian Edit distance computed with unmatched and matched symbol applied to alldata. This corresponds to the MKL results displayed in Table 5.16

Kernel no:	Data Size			
	158	134	115	104
1	0.4936	0.6322	0.1122	0.4522
2-3240	> 0.05	> 0.05	> 0.05	> 0.05
Mean	0.0003	0.0003	0.0003	0.0003
Standard Deviation	0.0087	0.0111	0.0021	0.0080

Table 5.20: Corresponding learned MKL weights for results displayed on Table 5.18

### 5.3.5 Multiple kernel learning of heterogeneous entities:

Multi-kernel learning with kernels derived from entities of diverse physical characteristics presents one of the key benefits of the kernel framework. As previously stated, the results reported in Tables 5.10, 5.14, 5.15, and 5.16 were based on homogeneous symbolic sequences, whereas the results reported in Tables 5.18 and 5.19 were based on numeric test measurements. These next series of experiments

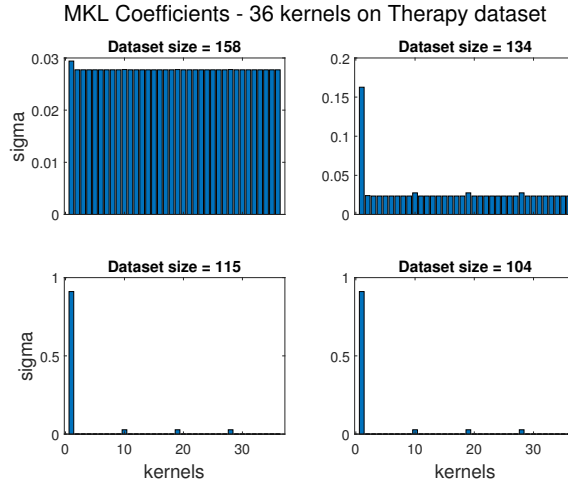


Figure 5.10: MKL coefficients learned from combining 36 Edit kernels applied to Therapy dataset. This corresponds to the MKL results displayed in Table 5.17

Kernel no:	Data Size			
	158	134	115	104
1	0.0000	0.0000	0.0000	0.0000
2-726	0.0012	0.0012	0.0012	0.0012
727-729	0.0013	0.0013	0.0013	0.0013
730-810	0.0012	0.0012	0.0012	0.0012

Table 5.21: Corresponding learned MKL weights for results displayed on Table 5.19

allowed us to apply MKL to combine symbolic and real-valued entities in a single classification model. We selected 11 out of 30 test measurements (listed in Table 4.6) based on having a higher distribution of events. Kernels via the 'edit distance on real sequence' ( $d_{ed11}$ ) were applied to the numeric entities and combined with the other kernels derived from symbolic data. The experiment specifically tested the classification performance of a heterogeneous combination of single test entities and symbolic data, with the goal of establishing the specific test measure as a separate entity that contributed to improving predictive capability. Concurrently, the experiment also identified those that degraded predictive performance. Afterwards, kernels derived from the entire set of 11 test measurements were combined with kernels generated from symbolic data.

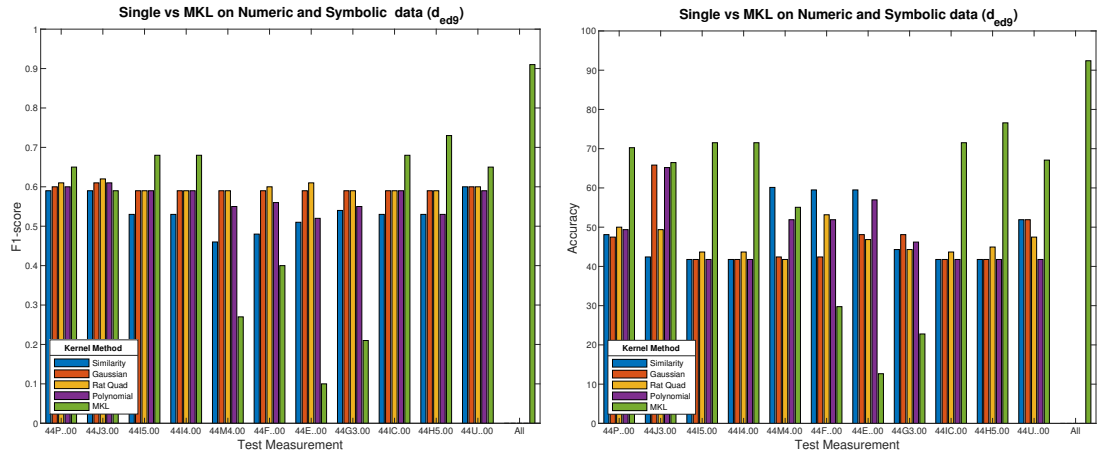


Figure 5.11: Single kernel performance vs MKL with Heterogeneous combination of edit kernels applied to 11 numeric Test measurements and edit kernel with distance ( $d_{ed9}$ ) applied to single view data (All data)

In the first experiment, we generated kernels via the 'edit distance computed with the number of unmatched and matched symbols' ( $d_{ed9}$ ) via 5 similarity edit kernels, 55 Gaussian edit kernels, 70 rational quadratic edit kernels, and 15 polynomial edit kernels. These are applied to the single view symbolic dataset (all data). For each real-valued test measure, 55 Gaussian edit kernels, 55 rational quadratic edit kernels, 5 similarity, and 15 polynomial edit kernels via the 'edit distance on real sequence' ( $d_{ed11}$ ) were generated. In total, for each test entity, we combined 130 kernels from real-valued test measurements with 145 from symbolic data. The F1-score and accuracy obtained for each test measure are displayed in Figure 5.11. The plot compares the best result obtained with stand-alone single kernels via the construction methods applied to the real-valued test measure against the MKL performance achieved by combining kernels (275) derived from the numeric test measure and symbolic data. A higher MKL F1-score was achieved in 6 (read codes, 44P.00, 44I5.00, 44I4.00, 44IC.00, and 44U.00) out of the 11 real-valued test measurements. It is also apparent from the plot that a very high performance was achieved by combining all kernels from the entire 11 test measures with kernels from symbolic data (1575). The MKL weight coefficient in Figure 5.12 shows a sparse solution was achieved for all test measures.

MKL Coefficients on Numeric and Symbolic data ( $d_{ed9}$ )

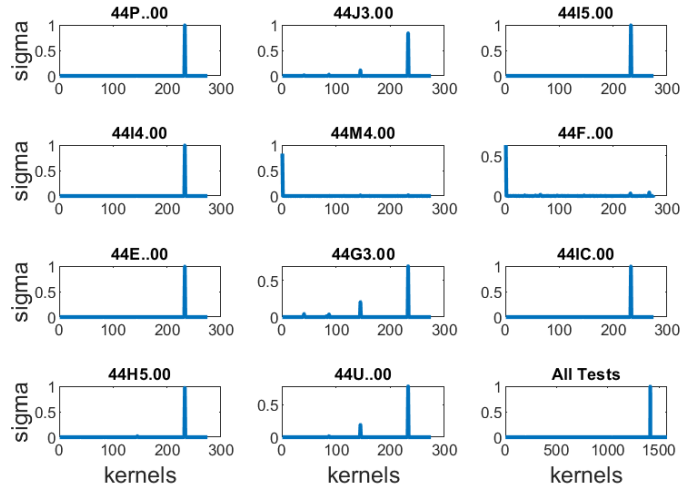


Figure 5.12: Distribution of MKL weight coefficients indicating SimpleMKL favours a sparse solution. These are the corresponding weights for the results in Figure 5.11

Similarly, in a second experiment, we combined 130 kernels generated for each test measure with 72 kernels derived via the distance ( $d_{ed10}$ ) made up of 12 polynomial edit kernels, 12 rational quadratic edit kernels, 44 Gaussian edit kernels, and 4 similarity edit kernels applied to the symbolic Therapy dataset. The MKL classification performance with 202 kernels only improved in 5 test measurements (44P.00, 44I4.00, 44G3.00, 44H5.00, and 44U.00). Refer to Figure 5.13. As evident in the plots, combining kernels from all test measurements with kernels derived from the symbolic Therapy dataset resulted in below-par classification performance. The learned MKL weight coefficients  $\sigma$  are displayed in Figure 5.14, which shows a sparse solution was achieved in all cases.

In a third experiment, symbolic kernels from experiments 1 (145 kernels) and 2 (72 kernels) were combined with additional (48) kernels generated from a symbolic single-view dataset. These new additions were made up of 12 edit polynomial kernels, 12 rational quadratic edit kernels, 20 Gaussian edit kernels, and 4 similarity edit kernels. A total of 265 symbolic kernels were combined with 130 kernels from real-valued test measurement, making it a total of 395 kernels per test measure.

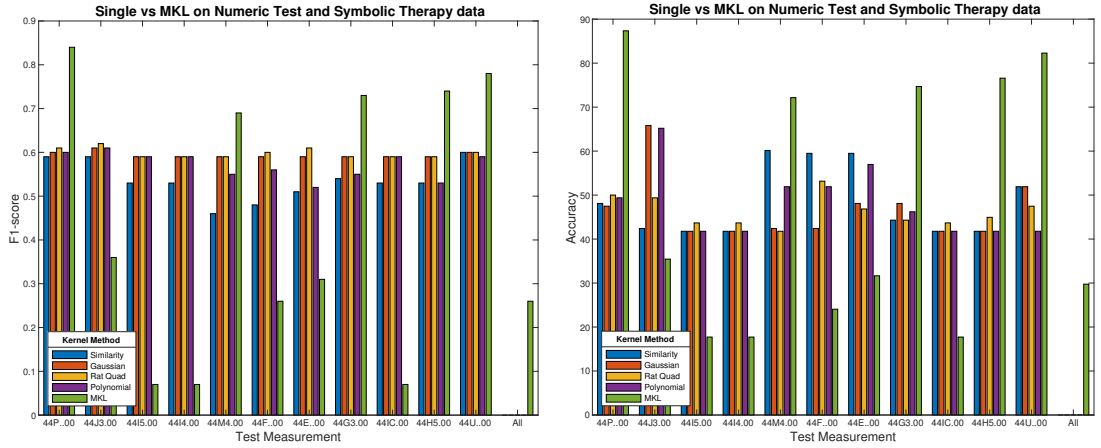


Figure 5.13: Best stand-alone single kernel performance vs MKL with heterogeneous combination of edit kernels applied 11 real-valued Test measurements and symbolic therapy dataset.

### MKL Coefficients on Numeric and Symbolic Therapy data

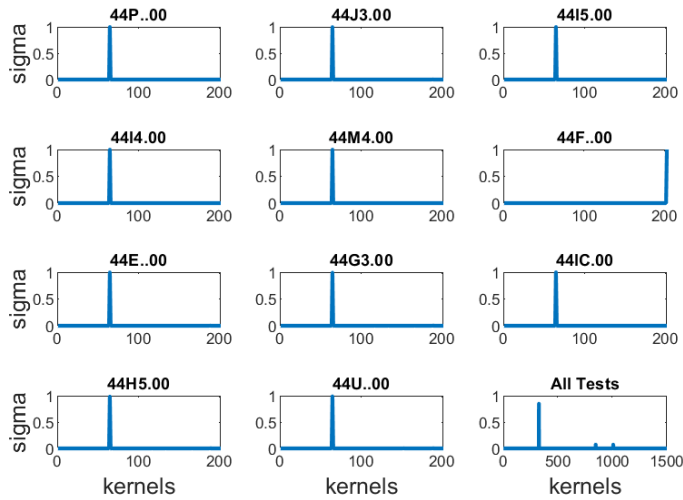


Figure 5.14: Distribution of MKL weight coefficients indicating SimpleMKL favours a sparse solution. These are the corresponding weights for the results in Figure 5.13

Lastly, kernels from all test measures (1430) were combined with the kernels from symbolic data (a total of 1695 kernels). The plot in Figure 5.15 shows the heterogeneous combination improved the classification result in 6 Tests: 44J3.00, 44I5.00, 44I4.00, 44E..00, 44G3.00, and 44IC.00. As can be seen, a high classification performance was also achieved with all test measurements. Figure 5.16 shows the corresponding learned MKL weights,  $\sigma$ . The comparison of the MKL heterogeneous combination of single real-valued test measurements with symbolic data for all three experiments is depicted in Figure 5.17. It also compares the results achieved with each single test addition to those achieved with all 11 test measurements. As can be seen from the plot, heterogeneous combinations with all test measures had a high classification performance in two out of the three experiments.

### 5.3.6 Measure predictive performance of the data tables

The predictive performance of the datasets was assessed based on the results obtained from meeting other experimental objectives, since the datasets offered a variety of things to experiment with. Consequently, the results from testing *Effects of different kernel functions* in subsection 5.3.1 were expressed in terms of the datasets. The best performance in Table 5.1 was obtained with the pseudo



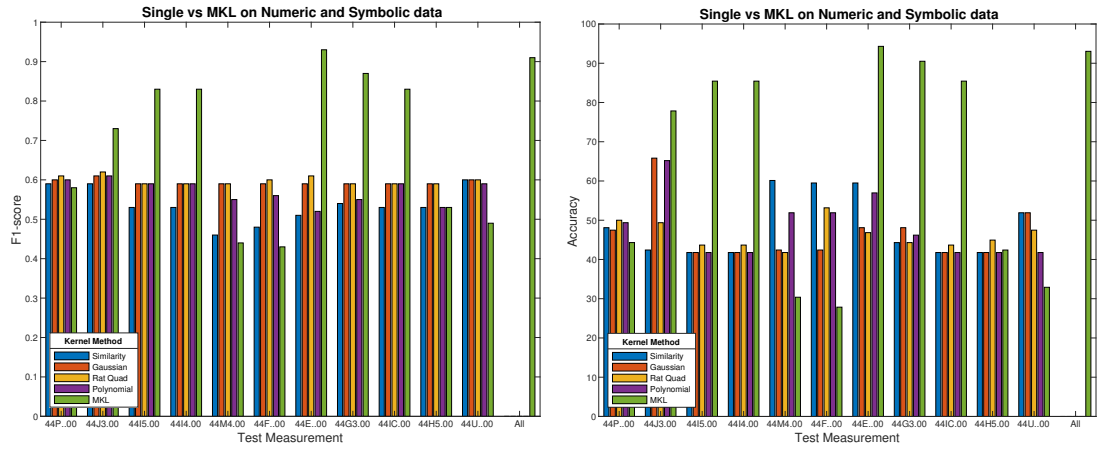


Figure 5.15: Single kernel performance vs MKL with Heterogeneous combination of edit kernels applied 11 numeric Test measurements and symbolic single view data using multiple edit distances.

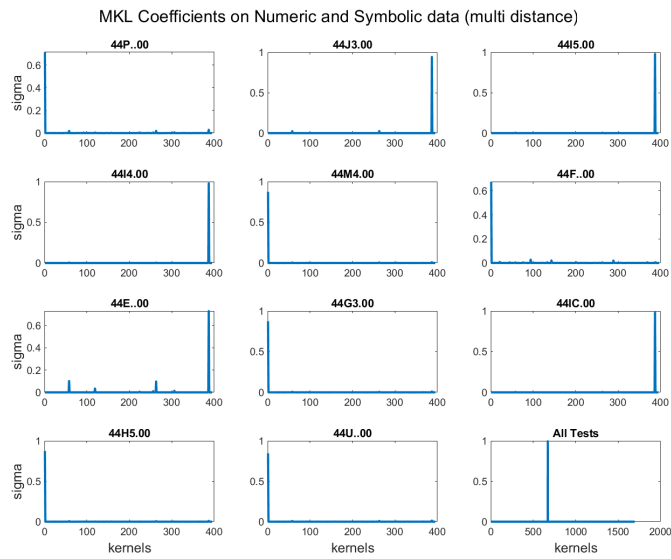


Figure 5.16: Distribution of MKL weight coefficients indicating SimpleMKL favours a sparse solution

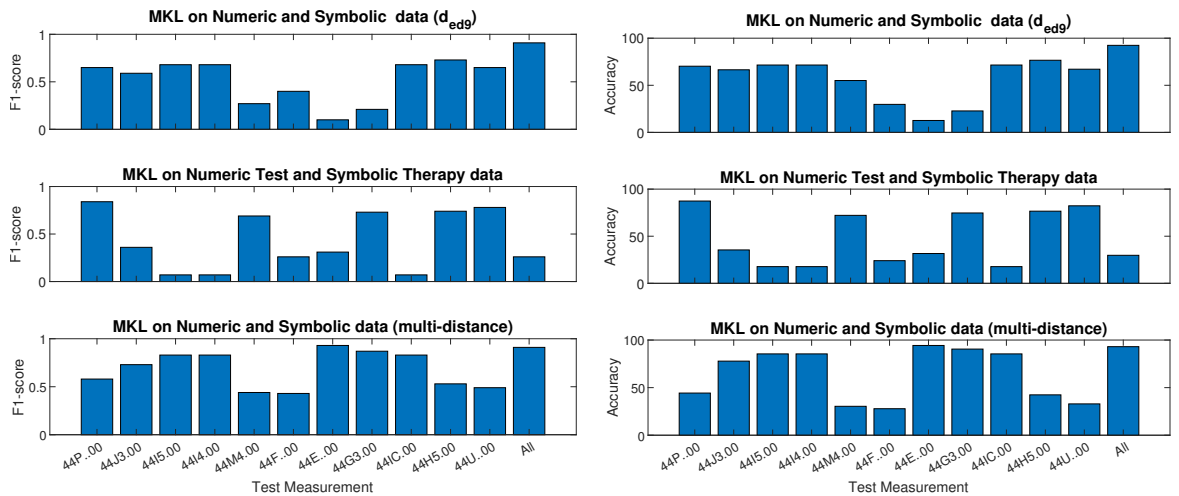


Figure 5.17: Comparison of F1-score and Accuracy achieved with 3 MKL experiments conducted based on heterogeneous combination of real-valued and symbolic data. The results for each single test is displayed in addition to the results achieved with all 11 test measurements

kernel via “edit distance normalized by length of longer sequence” for the Repeat dataset. However, the best results achieved with the similarity, Gaussian, rational quadratic, and polynomial edit kernels via ‘edit distance normalised by length of longer sequence’ in Tables 5.2, 5.3, 5.4, 5.5 were achieved with the Refer dataset. With the clinical dataset, the distance substitution method in Table 5.7 produced the best results. In the *Single vs. Multiple kernels* experiment in Subsection 5.3.3, the best result was achieved with the single view dataset (all data), as displayed in Table 5.10. Lastly, in the *Edit distance on variable-length sequences* experiment in Subsection 5.3.4, comparable optimum performance was obtained with the Repeat and Test dataset in Table 5.14

### 5.3.7 Static evaluation of kernels

In this section, we present the kernel goodness metrics applied to assess the quality of the generated kernels. These static measures can give an indication of the quality of pattern representation in the feature space and are applied in *kernel learning* from data. Given a set of kernels generated via a set of distances and applied to the datasets, the number of PSD kernels, the number of indefinite kernels (NSD), the mean percentage number of negative eigenvalues obtained with the indefinite kernels, the mean spectral ratio, and the mean kernel target alignment (KTA) are captured. The goodness measures for the kernels generated using the distance substitution method are displayed in Table 5.22. Each row represents the measurements obtained with 158 kernels generated with the specified distance function.

Dataset	Distance	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
<b>Clinical</b>	$k_{ds\_ed1}$	158	0	0.00 (0.00)	2.16 (0.54)	0.95 (0.06)
	$k_{ds\_ed2}$	158	0	0.00 (0.00)	2.41 (0.90)	0.93 (0.05)
	$k_{ds\_ed3}$	0	158	22.80 (7.40)	0.91 (0.59)	0.74 (0.09)
	$k_{ds\_ed4}$	0	158	42.82 (9.13)	0.33 (0.41)	0.49 (0.14)
<b>Recall</b>	$k_{ds\_ed1}$	0	158	23.56 (0.72)	1.69 (0.26)	0.74 (0.11)
	$k_{ds\_ed2}$	0	158	17.56 (4.27)	1.92 (0.40)	0.77 (0.10)
	$k_{ds\_ed3}$	0	158	27.60 (3.72)	1.53 (0.29)	0.73 (0.11)
	$k_{ds\_ed4}$	0	158	28.44 (3.97)	1.50 (0.30)	0.72 (0.10)
<b>Refer</b>	$k_{ds\_ed1}$	0	158	3.52 (0.63)	2.25 (0.74)	0.91 (0.13)
	$k_{ds\_ed2}$	0	158	3.20 (0.73)	2.43 (1.05)	0.88 (0.12)
	$k_{ds\_ed3}$	0	158	4.82 (1.88)	2.22 (0.75)	0.91 (0.13)
	$k_{ds\_ed4}$	0	158	8.62 (3.24)	1.98 (0.82)	0.89 (0.12)
<b>Repeat</b>	$k_{ds\_ed1}$	82	76	0.64 (0.00)	2.63 (1.35)	0.96 (0.09)
	$k_{ds\_ed2}$	75	83	0.64 (0.00)	2.66 (1.55)	0.90 (0.10)
	$k_{ds\_ed3}$	16	142	7.13 (6.16)	2.43 (1.43)	0.96 (0.09)
	$k_{ds\_ed4}$	7	151	10.93 (6.36)	2.09 (1.46)	0.94 (0.08)
<b>Test</b>	$k_{ds\_ed1}$	0	158	17.13 (0.66)	1.65 (0.21)	0.75 (0.14)
	$k_{ds\_ed2}$	0	158	11.81 (4.21)	2.05 (0.63)	0.77 (0.09)
	$k_{ds\_ed3}$	0	158	30.73 (7.07)	1.05 (0.41)	0.66 (0.07)
	$k_{ds\_ed4}$	0	158	39.34 (12.10)	0.90 (0.45)	0.61 (0.06)
<b>Therapy</b>	$k_{ds\_ed1}$	158	0	0.00 (0.00)	3.14 (1.87)	0.91 (0.15)

Distance Substitution kernel Evaluation

Table 5.22 –

Dataset	Distance	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
<b>All data</b>	$k_{ds\_ed2}$	158	0	0.00 (0.00)	2.90 (1.91)	0.84 (0.15)
	$k_{ds\_ed3}$	5	153	26.35 (13.72)	2.04 (1.90)	0.90 (0.13)
	$k_{ds\_ed4}$	0	158	26.78 (10.18)	1.48 (1.67)	0.86 (0.10)
	$k_{ds\_ed1}$	158	0	0.00 (0.00)	2.48 (0.98)	0.95 (0.07)
	$k_{ds\_ed2}$	158	0	0.00 (0.00)	2.59 (1.22)	0.90 (0.08)
	$k_{ds\_ed3}$	0	158	26.45 (8.79)	0.91 (0.81)	0.71 (0.10)
	$k_{ds\_ed4}$	0	158	46.24 (6.00)	0.19 (0.31)	0.38 (0.15)

Table 5.22: Distance substitution kernel assessment

In Table 5.23, the abridged version of the kernel goodness measures that correspond to the kernels generated in Section 5.3.1 are displayed. The table shows the measures according to the kernel construction method and dataset. Tables C.1, C.2, C.3, C.4, C.5, C.6, and C.7 of Appendix C provide a much more detailed presentation of the goodness measure. Likewise, in Table 5.24 the abridged version of the kernel goodness measure obtained with the experiment - *Edit distance on sequences with common items* in Section 5.3.2 is displayed, while the full version is in Table C.8 of Appendix C.

Dataset	Method	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
<b>Clinical</b>	Pseudo	0	4	75.16 (25.65)	0.00 (0.00)	0.74 (0.32)
	Similarity	2	2	25.63 (33.57)	6.35 (5.53)	0.67 (0.38)
	Gaussian	8	12	31.59 (20.97)	4.65 (5.22)	0.70 (0.41)
	Rat Quad	7	5	29.49 (25.02)	7.38 (5.23)	0.56 (0.37)
	Poly	6	6	25.32 (25.89)	4.60 (3.63)	0.87 (0.18)
<b>Recall</b>	Pseudo	0	4	68.99 (12.64)	0.00 (0.00)	0.83 (0.08)
	Similarity	0	4	7.91 (1.51)	4.93 (2.41)	0.73 (0.19)
	Gaussian	0	20	21.11 (13.95)	3.97 (3.89)	0.69 (0.41)
	Rat Quad	0	12	7.65 (1.51)	6.31 (2.93)	0.57 (0.30)
	Poly	0	12	7.91 (0.99)	4.56 (2.50)	0.84 (0.19)
<b>Refer</b>	Pseudo	0	4	81.65 (14.90)	0.00 (0.00)	0.90 (0.05)
	Similarity	0	4	3.16 (0.90)	5.08 (2.00)	0.81 (0.12)
	Gaussian	0	20	11.93 (12.68)	4.38 (4.40)	0.69 (0.42)
	Rat Quad	0	12	3.32 (1.12)	6.72 (3.18)	0.61 (0.33)
	Poly	0	12	3.22 (1.28)	4.94 (2.56)	0.86 (0.17)
<b>Repeat</b>	Pseudo	0	4	76.27 (23.93)	0.00 (0.00)	0.90 (0.06)
	Similarity	2	2	2.85 (0.45)	8.36 (4.14)	0.60 (0.26)
	Gaussian	5	15	18.23 (20.58)	5.36 (5.32)	0.66 (0.43)
	Rat Quad	2	10	1.65 (1.41)	9.54 (3.69)	0.46 (0.30)
	Poly	6	6	1.90 (1.06)	6.01 (3.39)	0.84 (0.21)
<b>Test</b>	Pseudo	0	4	69.30 (16.77)	0.00 (0.00)	0.75 (0.20)

Kernel Evaluation Summary

Table 5.23 –

Dataset	Method	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
<b>Therapy</b>	Similarity	0	4	15.35 (23.18)	4.19 (3.16)	0.68 (0.26)
	Gaussian	0	20	21.46 (17.36)	3.53 (3.84)	0.71 (0.36)
	Rat Quad	0	12	14.50 (18.87)	5.44 (3.79)	0.56 (0.29)
	Poly	0	12	14.45 (19.16)	3.88 (2.65)	0.82 (0.17)
	Pseudo	0	4	72.31 (27.74)	0.00 (0.00)	0.80 (0.15)
	Similarity	2	2	11.08 (11.19)	8.05 (5.26)	0.41 (0.40)
	Gaussian	8	12	24.89 (22.47)	5.76 (4.97)	0.57 (0.41)
	Rat Quad	7	5	10.13 (8.44)	9.24 (4.41)	0.33 (0.33)
	Poly	6	6	8.86 (7.22)	5.60 (3.22)	0.80 (0.23)
	<b>All data</b>	Pseudo	0	4	75.79 (24.99)	0.00 (0.00)
Similarity		2	2	26.58 (34.01)	6.84 (5.97)	0.60 (0.43)
Gaussian		8	12	29.85 (21.69)	4.95 (5.21)	0.67 (0.41)
Rat Quad		7	5	31.14 (26.12)	7.72 (5.46)	0.52 (0.39)
Poly		6	6	25.95 (26.35)	4.70 (3.72)	0.86 (0.19)

Table 5.23: Static kernel assessment Summary

In Table 5.25, the goodness measure achieved with the template matching kernel construction method of Table 5.6 is displayed. In Table 5.26 the goodness measures for the kernels generated in the experiment *Multiple kernel learning of heterogeneous entities* in section 5.3.5 are displayed. The measures in Tables 5.27 and 5.28 correspond to the kernels with classification performance results in Tables 5.16 and 5.17 respectively.

Dataset	Method	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
<b>All data</b>	Pseudo	0	4	61.39 (8.88)	0.00 (0.00)	0.72 (0.23)
	Similarity	0	4	21.99 (20.08)	4.05 (3.25)	0.76 (0.29)
	Gaussian	0	44	29.59 (16.54)	3.23 (3.46)	0.75 (0.36)
	Rat Quad	0	56	21.55 (18.61)	5.23 (4.04)	0.55 (0.38)
	Poly	0	12	19.15 (19.41)	3.68 (2.58)	0.86 (0.17)

Table 5.24: Kernel goodness measure for kernels generated from Edit distance on sequences with common items in Figure 5.4

Dataset	Method	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
<b>All data</b>	Temp Match	0	158	49.54 (0.55)	1.00 (0.00)	0.85 (0.09)

Table 5.25: Kernel goodness measure for kernels generated using the template matching method with results displayed in Table 5.6

### Spectral modification result comparison

For completeness, a visual illustration of the comparison of the results obtained with the spectrum modification applied to indefinite kernels in some of the experiments is plotted. As indicated in previous sections, the highest F1-score achieved is reported with an indication of the modification, if any, that was applied. The Figure 5.20 show the comparison of the F1-scores obtained with the spectral transformations (clip, shift, flip, and square). This corresponds to the experiment, the results

Dataset	Method	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
<b>Test</b>	Pseudo	0	55	62.19 (8.43)	0.00 (0.00)	0.80 (0.04)
	Similarity	0	55	29.46 (11.53)	2.05 (0.65)	0.81 (0.04)
	Gaussian	0	605	33.94 (10.51)	1.57 (1.11)	0.83 (0.25)
	Rat Quad	0	605	30.77 (11.61)	2.20 (1.45)	0.62 (0.24)
	Poly	0	165	29.81 (11.41)	2.08 (0.84)	0.79 (0.12)

Table 5.26: Kernel goodness measure for kernels generated multiple kernel learning of heterogeneous entities in section 5.3.5. The kernels implement the distance  $d_{ed10}$

Dataset	Method	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
<b>All data</b>	Pseudo	0	5	48.99 (1.31)	0.00 (0.00)	0.01 (0.02)
	Similarity	0	5	31.52 (20.49)	3.23 (2.55)	0.55 (0.39)
	Gaussian	0	55	32.32 (18.08)	3.11 (2.23)	0.56 (0.36)
	Rat Quad	0	70	31.11 (18.57)	3.32 (2.42)	0.54 (0.35)
	Poly	0	15	31.52 (19.02)	2.71 (1.86)	0.78 (0.22)

Table 5.27: Kernel goodness measure for kernels with results displayed in Table 5.16

Dataset	Method	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
<b>Therapy</b>	Pseudo	0	4	99.37 (0.00)	0.00 (0.00)	0.84 (0.00)
	Similarity	4	0	0.00 (0.00)	12.53 (0.00)	0.12 (0.00)
	Gaussian	44	0	0.00 (0.00)	7.80 (5.09)	0.42 (0.39)
	Rat Quad	12	0	0.00 (0.00)	12.54 (0.02)	0.11 (0.02)
	Poly	12	0	0.00 (0.00)	7.40 (3.40)	0.77 (0.24)

Table 5.28: Kernel goodness measure for kernels with results displayed in Table 5.17

of which are shown in Table 5.15 for a dataset size of 158. Likewise, the Figure 5.21 correspond to the results obtained and displayed in Table 5.16. It is interesting to note from the plot in Figure 5.22 that only 4 indefinite kernels from the experiment in Table 5.17 (Size 158) were modified.

### Zero vector candidates

In the Figures 5.18 and 5.19, the plots of the F1-scores achieved with each data point applied as the ‘zero vector’ in the distance substitution kernels that were combined with MKL are displayed. The actual results showing the best candidates are displayed in Table 5.10. It is striking to note that the candidate with the best F1-score stood out in comparison to others in the majority of the datasets.

### 5.3.8 Comparison with traditional bag-of-words (BOW)

<i>Models</i>		<i>Bag-of-Words</i>				<i>Binary Bag-of-Words</i>			
		<b>F1</b>	<b>Acc(%) <math>\pm</math> (std)</b>	<b>Sen</b>	<b>Spec</b>	<b>F1</b>	<b>Acc(%) <math>\pm</math> (std)</b>	<b>Sen</b>	<b>Spec</b>
<b>SVM</b>	<b>Linear</b>	0.51	56.96 (50.00)	0.49	0.64	0.61	67.09 (47.00)	0.61	0.72
	<b>Poly</b>	0.62	62.03 (49.00)	0.53	0.74	0.59	67.09 (47.00)	0.61	0.71
	<b>RBF</b>	0.25	62.03 (49.00)	0.71	0.61	0.25	62.03 (49.00)	0.61	0.71
	<b>Exp RBF</b>	0.20	60.76 (49.00)	0.67	0.60	0.18	57.59 (49.00)	0.47	0.59
	<b>Laplace</b>	<b>0.63</b>	<b>69.62 (46.00)</b>	<b>0.65</b>	<b>0.73</b>	0.62	68.99 (46.00)	0.63	0.73
<b>Logistic Regression</b>		0.51	56.96 (50.00)	0.49	0.64	<b>0.63</b>	<b>70.25 (45.00)</b>	<b>0.65</b>	<b>0.74</b>

Table 5.29: Performance result obtained with Logistic Regression and SVM applied to bag-of-words and binary bag-of-words feature representation of the data

BoW feature representation is one of the traditional methods commonly used to represent sequential data in vectorial form. As a result, the method is employed in order to extract vector space features from the symbolic (all data) dataset. Logistic regression, SVM, and deep learning multi-layer

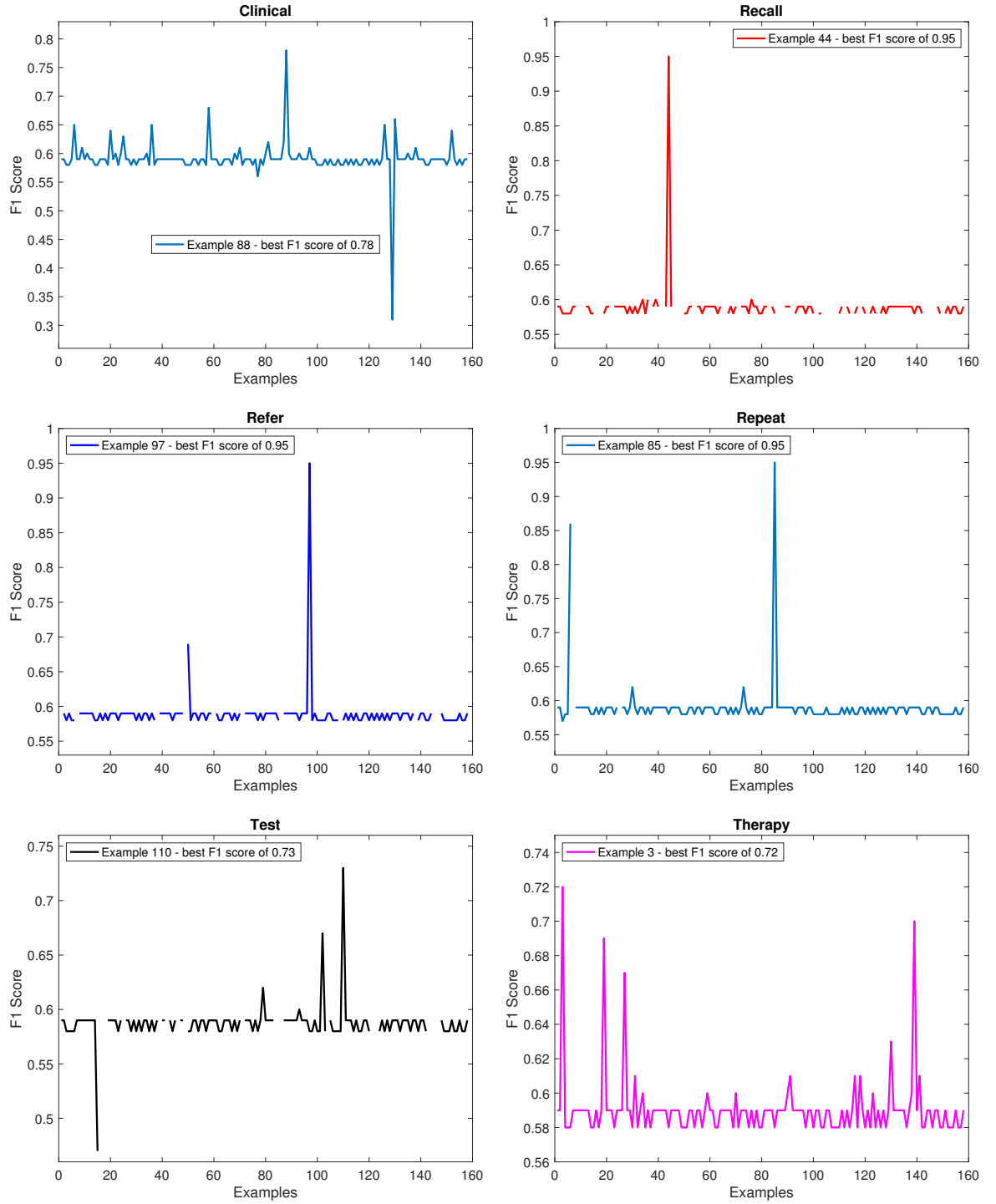


Figure 5.18: F1-Scores obtained with each example used as the zero vector in the MKL combination of kernels generated with the distance substitution kernel construction method applied to the six datasets. These correspond to the results displayed in Table 5.10

Models		Bag-of-Words				Binary Bag-of-Words			
		F1	Acc(%)	Sen	Spec	F1	Acc(%)	Sen	Spec
Deep Learning	MLP	0.54	60.76 (48.98)	0.53	0.67	0.46	59.49 (49.25)	0.52	0.63
	LSTM	0.44	51.90 (50.12)	0.43	0.59	<b>0.61</b>	<b>51.27 (50.14)</b>	<b>0.46</b>	<b>0.80</b>

Table 5.30: Performance result obtained with deep learning LSTM and MLP applied to bag-of-words and binary bag-of-words feature representation of the data

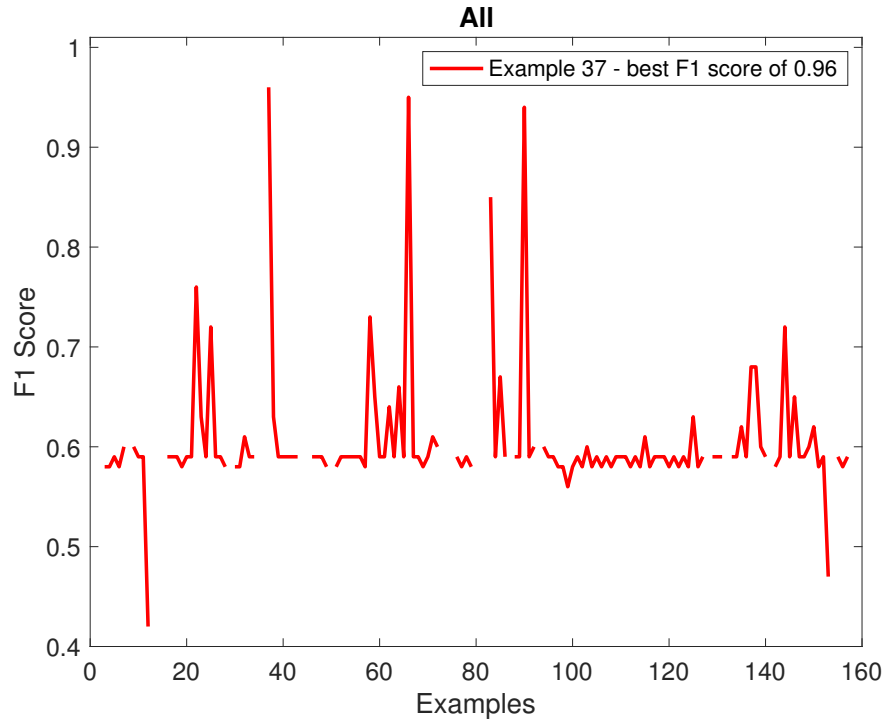


Figure 5.19: F1-Scores obtained with each example used as the zero vector in the MKL combination of kernels generated with the distance substitution kernel construction method applied to the ‘All data’ dataset. These correspond to the results displayed in Table 5.10

perceptron (MLP) and long short-term memory (LSTM) are applied to the BoW features as a baseline to compare against the proposed model. Table 5.29 and 5.30 show the results obtained. The SVM is tested with the standard kernel functions: linear, polynomial, RBF, exponential RBF and Laplace, which work well with even-length vectorized data. Experimenting with the same leave-one-out cross-validation (LOOCV), the F1-score of **0.63** and accuracy of **69.62** was achieved with SVM implementing the Laplace kernel on the bag-of-words features. The deep learning models with results displayed on Table 5.30 were also partitioned using the LOOCV approach. The LSTM on the binary bag-of-words outperformed the LSTM on regular bag-of-words features with an F1-score of **0.61** and accuracy of **51.27**

### 5.3.9 Validation on external data

The results of the experiment on external validation data implemented via the distance substitution method are displayed in Table 5.31, which shows the kernel function via ‘edit distance normalized by the length of the longer sequence’ ( $\mathbf{k}_{ds\_ed2}$ ) achieved the best F1-score of **0.95** and **95%** accuracy. The kernel via native ‘edit distance’ ( $\mathbf{k}_{ds\_ed1}$ ) and ‘edit distance normalized by the number of common items’ ( $\mathbf{k}_{ds\_ed3}$ ) both achieved the same F1-score of **0.93** with **92.50%** and **93.75%** accuracy, respectively. Using “edit distance normalised by exponent of number of common items” ( $\mathbf{k}_{ds\_ed4}$ ), the kernel achieved an F1-score of **0.90** and **0.95%** accuracy

The results obtained from MKL applied to the validation dataset are displayed on Table 5.32. The experiment combined the best performing kernels and obtained an F1-score of **0.97** and an accuracy of **96.75%**. In addition, the validation dataset was transformed into feature vectors with the bag-of-words representations, and the results are displayed in Tables 5.32 and 5.34. On the bag-of-words features, SVM implemented with the Laplace kernel yielded an F1 score of **0.95** and an accuracy of **96.52%**. The results of the deep learning experiments on the validation dataset are displayed in Table 5.33. It demonstrates that the LSTM model obtained an F1 score of **0.87%** and

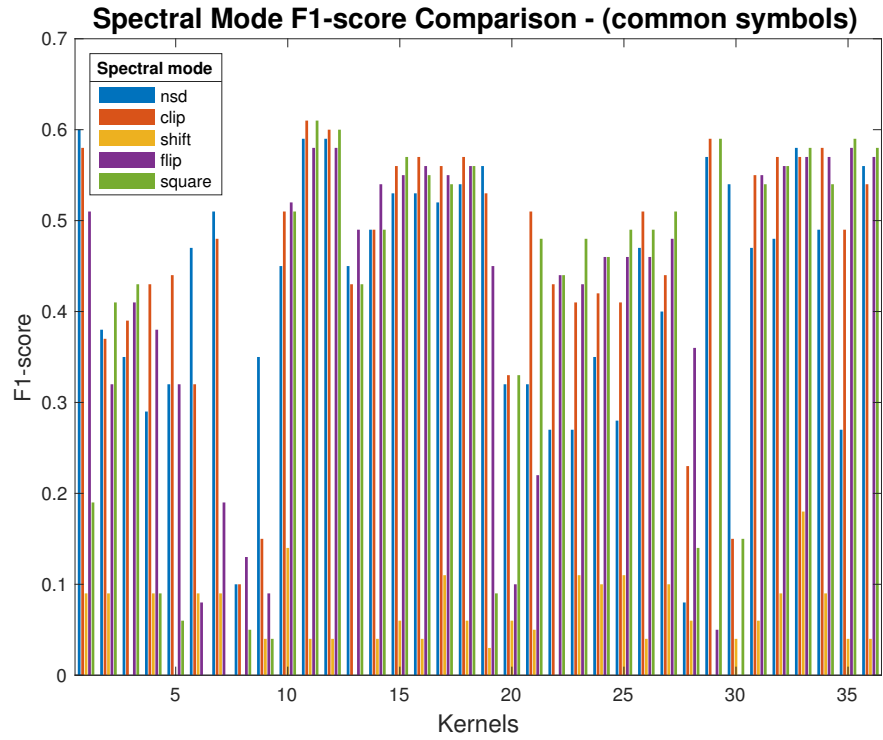


Figure 5.20: F1 score obtained by spectral transformation (clip, shift, flip and square) compared against indefinite kernels of the 36 Gaussian Edit kernels (common symbols) and applied to the single view dataset.

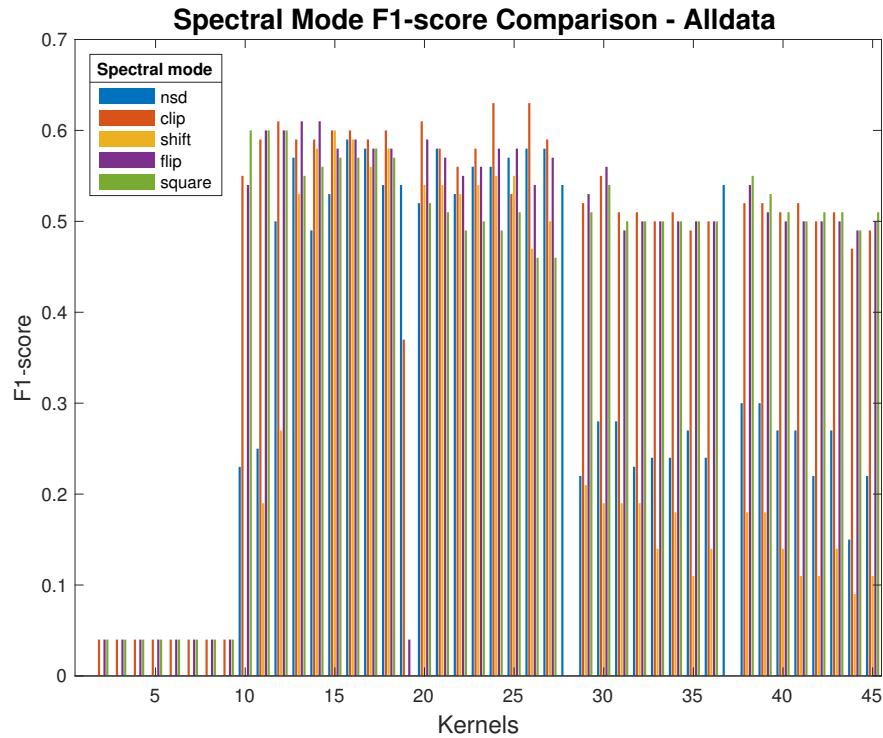


Figure 5.21: F1 score obtained by spectral transformation (clip, shift, flip and square) compared against indefinite kernels of the 45 Gaussian Edit kernels (distance computed with matched and unmatched symbols) applied to the single view dataset.

an accuracy of **85.05%** percent.



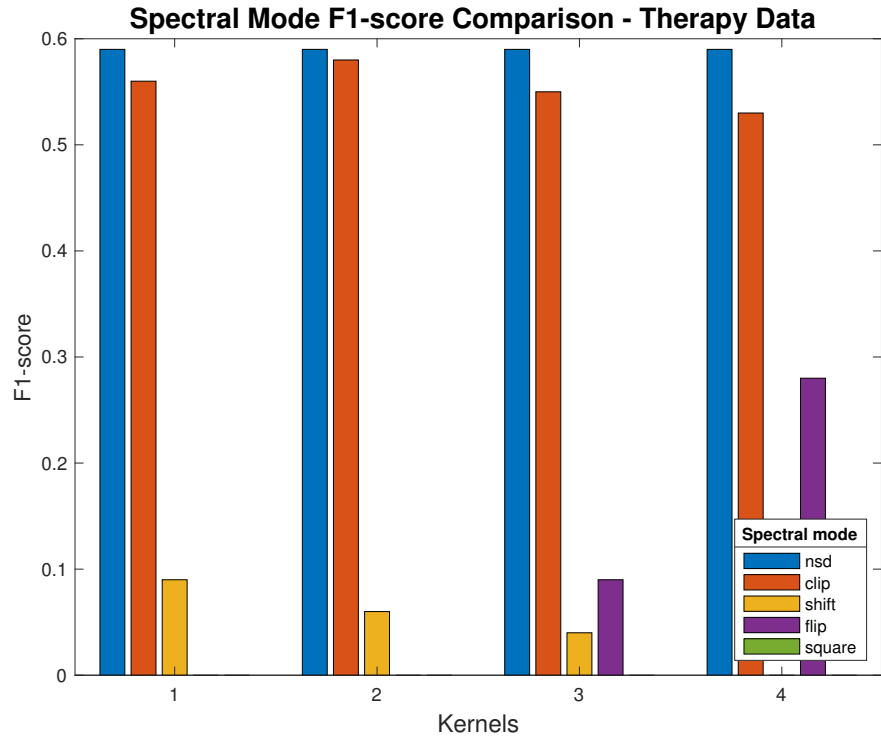


Figure 5.22: F1 score obtained by spectral transformation (clip, shift, flip and square) compared against 4 indefinite kernels out of the 36 Gaussian Edit kernels applied to the Therapy dataset.

Kernel	F1	Acc(%)	Sen	Spec	nSv	%-Eig
$k_{ds\_ed1}$	0.90	90.58	0.98	0.85	0	156.35
$k_{ds\_ed2}$	0.87	85.71	0.79	0.97	0	156.35
$k_{ds\_ed3}$	<b>0.93</b>	<b>92.86</b>	<b>0.93</b>	<b>0.93</b>	<b>0</b>	<b>156.35</b>
$k_{ds\_ed4}$	0.92	90.56	0.91	0.93	0	156.35

Table 5.31: Best results obtained from classification with single kernels applied to the validation (peptide) data

Kernels	F1	Acc(%)	Sen	Spec
<b>Top 7 Kernels</b>	0.97	96.75	1.00	0.93

Table 5.32: Best MKL results obtained by combining the top 7 kernels applied to the validation (peptide) data

Models		Bag-of-Words				Binary Bag-of-Words			
		F1	Acc(%)	Sen	Spec	F1	Acc(%)	Sen	Spec
Deep Learning	MLP	0.54	60.76 (48.98)	0.53	0.67	0.46	59.49 (49.25)	0.89	0.63
	LSTM	0.83	84.54 (36.20)	0.80	0.87	<b>0.87</b>	<b>85.05 (35.70)</b>	<b>0.79</b>	<b>0.95</b>

Table 5.33: Performance result obtained with deep learning LSTM and MLP applied to bag-of-words and Binary bag-of-words feature representation of the validation (peptide) data

## 5.4 Summary

The results in this chapter indicate that the kernel framework provides a principled way of addressing the issues with predictive modelling of irregularly sampled, arbitrary-length heterogeneous EHR data. It shows the several methods that can be applied to convert an elastic distance measure into a valid kernel function. The results of the experiment with varying sequence lengths show that the edit distance is a suitable method of addressing uneven length sequences. Where necessary, the spectrum of indefinite kernels may be modified in addition to applying post-kernel processing in order to enhance the possibilities of finding patterns in the induced feature space. Poor performing stand-alone kernels

<i>Models</i>		<i>Bag-of-Words</i>				<i>Binary Bag-of-Words</i>			
		<b>F1</b>	<b>Acc(%) <math>\pm</math> (std)</b>	<b>Sen</b>	<b>Spec</b>	<b>F1</b>	<b>Acc(%) <math>\pm</math> (std)</b>	<b>Sen</b>	<b>Spec</b>
<b>SVM</b>	<b>Linear</b>	0.92	92.01(27.11)	0.93	0.91	0.91	91.49 (28.00)	0.92	0.91
	<b>Poly</b>	0.93	93.56 (25.00)	0.93	0.94	0.92	92.27 (27.00)	0.94	0.91
	<b>RBF</b>	0.92	92.27 (27.00)	0.89	0.96	0.92	92.01 (27.00)	0.90	0.94
	<b>Exp RBF</b>	0.91	90.21 (30.00)	0.85	0.98	0.91	90.27 (30.00)	0.85	0.97
	<b>Laplace</b>	<b>0.95</b>	<b>95.36 (21.00)</b>	<b>0.94</b>	<b>0.96</b>	0.92	91.75 (28.00)	0.91	0.93
<b>Logistic Regression</b>		0.92	92.01 (27.10)	0.92	0.92	0.92	92.53 (26.30)	0.92	0.93

Table 5.34: Performance result obtained with Logistic Regression and SVM applied to bag-of-words and Binary bag-of-words feature representation of the validation (peptide) data

may also be combined to achieve a better result. The utility of the framework is expressed by the use of MKL to combine kernels from diverse entities into a single model. In addition, the validation of the method on external symbolic data provides further proof of its appeal. Taken together, this chapter provides insight into the suitability of the kernel framework as a risk prognosis tool that can be applied in chronic disease management in the context of a primary care setting. The next chapter, therefore, moves on to discuss the findings.

# Chapter 6

## Discussion

The purpose of this study was to see how well an edit distance kernel-based machine learning framework performed in the predictive modelling of heterogeneous and irregularly sampled arbitrary-length EHR data. Such undertakings usually involve a strenuous feature extraction process with the potential for information loss in the representation used by standard machine learning approaches. This work specifically sought to address this shortcoming by adopting a featureless, discrete, symbolic representation of the EHR. This was necessary in order to utilise an NLP sequence analysis method like the elastic edit distance proximity measure. Its practicality as a viable option for addressing the highlighted problems is extended in three ways. Firstly, the longitudinal temporal order of the clinical events is preserved in calculating the proximity between objects. Prior studies, as highlighted in the literature [266] affirmed that this concern still persists with modelling with EHR. Secondly, it addressed the problem of irregularly sampled arbitrary length sequences. Lastly, as this research has demonstrated, it can be implemented in conjunction with the kernel framework in predictive modelling with EHR.

The study used a case study to develop a machine learning predictive prognosis model capable of identifying healthy patients at risk of developing type 2 diabetes based on an elevated blood pressure of 130/80 mm Hg. Our findings from the experiments show that the proposed EHR kernel framework implemented via edit distance-based kernels achieved high predictive performance even when there were significant disparities in EHR sequence size and sampling regularity. In addition, the results also showed that the problems of incorporating heterogeneous entities into a single model can be addressed in a principled manner with the multi-kernel learning model.

The first round of experiments is in agreement with the modularity of the process in the sense that we can decouple the data representation and pairwise similarity construct from actual kernel development. Consequently, the focus of the learning process shifts towards crafting variants of the edit distance measures that can address specific domain-centric problems. This provides further proof of the adaptability and usability of the kernel framework to address problems in other domains. As mentioned in the literature review, this study also showed that the edit distance can be tailored to address specific problems. By introducing distance normalization, controlling the equality of symbols, computing the ratio of matched and unmatched symbols, and subsequence edit distance, the native edit distance was extended to incorporate some form of human knowledge about the similarity of clinical and non-clinical items.

The results from the first experimental objective show these variants of edit distance measures, as defined in section 4.2.1, can be transformed into valid kernels using established standard kernel construction methods. The most interesting finding was that the ‘distance substitution’ method

outperformed the other methods. A possible explanation for this might be that the translation of the points into the origin of the feature space enhanced the linear separability of the objects, thus proving that the embedding of distance measures expressed in terms of norms in the feature space is an intuitive way of utilising the kernel framework with metric distance measures. What was surprising was that using a centring post-kernel transformation to move the origin of the feature space to the centre of the mass of the data did not improve the classification performance. Another important finding was that all kernel methods via the distances applied to the datasets, with the exception of the pseudo method, resulted in at least a few PSD kernels. However, the empirical findings show that PSD kernels did not guarantee optimum classification performance, as they were outperformed by indefinite kernels with modified spectra. Nevertheless, it showed an RKHS can be induced with edit kernels.

The second experimental objective, to test the effect of the variation of sequence lengths, showed no significant improvement for a closer gap between the lengths of a given pair of sequences. A possible explanation for the poor test result may be that the further reduction of an already small sample size proved insufficient to properly learn the target function, and as a result, the poor test result was seen. This objective would have been adequately examined with a larger cohort; thus, it is an important issue for future research. Nevertheless, empirical results confirm that we can apply MKL to combine poor base kernels to achieve a significantly better model in an EHR context, in contrast to the results obtained with traditional  $L_1$  MKL which are usually poor [261]. This finding gives strong support for kernel-based symbolic data representation as a suitable approach for modelling longitudinal clinical data. MKL of distance substitution kernels resulted in a relatively high number of support vectors (in terms of which the final decision boundary is described). This may suggest a potential for overfitting if too many kernels are used. In contrast, a smaller number of support vectors were seen in the majority of the MKL combination of Gaussian edit kernels.

One unanticipated finding was the perfect classification result achieved with the combination of several Gaussian ‘edit distance on real sequence’ kernels applied to the test measurements (Table 5.19). This result may be explained by a combination of factors. The small distribution of numeric test measurements recorded in some of the examples and the complex use of 810 composite kernels derived via the ‘edit distance with real sequence’ may have contributed to over-learning. The learned weight coefficient  $\sigma$  for all MKL processes showed convergence to a sparse solution, which is consistent with the SimpleMKL algorithm’s expectations.

It is interesting to note that the results showed that the proposed kernel framework provides a proper method of addressing heterogeneous clinical entities. Real-valued test measurements can be combined with the symbolic representation of longitudinal categorical healthcare data. The order of the entities that would have otherwise been discarded with standard methods was incorporated into the classification model; thus somewhat contributing to classifying the data. It also provides a means of filtering out test results that may contribute to the overall classification. When applied to kernels derived from the Therapy dataset, the full complement of the test results performed poorly, in contrast to kernels derived from the composite ‘single view’ data set that also includes the Therapy data. If all test results are used, the Therapy dataset as a standalone may not be ideal for increasing the chances of finding separability. We have also demonstrated the utility of the elastic edit distance proximity measure by its applicability to symbolic and real-valued time series.

The findings indicate that certain data modalities hold more intrinsic predictive value, with the clinical dataset having the best single (distance substitution) kernel performance, an indication that patient medical histories hold the most informative value regarding patient behaviour. However,

extracting all the data into a composite ‘single view’ dataset outperformed the result achieved by the clinical dataset alone; despite a similar F1-score of 0.74, the single view dataset achieves a higher sensitivity of 0.98. (Although its specificity of 0.51 means that it fails to identify half of those less likely to succumb to the disease, we nevertheless accept the outcome on the basis that the anticipated intervention of prescribing a healthy lifestyle to people at risk of the disease is not deemed harmful to healthy patients). This distinct difference in performance was only seen with the distance substitution method.

Some test results reflect routine medical procedures rather than actual health status. For instance, creatinine can be ordered for monitoring disease progression as well as for routine panels for preventive testing as stipulated by guidelines; thus, the context in which a laboratory test is ordered depends both on its clinical purpose and the surrounding healthcare processes [200]. Based on this knowledge, it is expected that some tests in this context may only contribute noise to the MKL process. The high predictive value of the F1-score of 0.91 seen when all 11 test measures were used shows the performance did not degrade by a great deal in comparison to the high value of 0.93 seen with MKL with the serum bilirubin level (44E.00) test measurement.

The greedy search for the ‘zero-vector’ candidate is one of the drawbacks seen with the template matching and distance substitution methods. The plot of the F1-scores obtained with each example used as the candidate showed a spike indicating the existence of the perfect candidate. With the exception of the therapy and ‘single view’ composite datasets, there was a significant gap between the performance obtained with the ideal candidate in comparison to the rest. Despite the good classification performance achieved, applying cross-validation to iterate through the entire dataset to find the optimum candidates makes it an undesirable choice if working with a large dataset.

In practice, it is difficult to establish a priori similarity measures that will yield the best classification result since such qualities are inherently data specific (in effect, the ‘no free lunch’ theorem). The edit distance computation is based on minimizing the weighted edit cost incurred in transforming one sequence into another. This, however, ignores any effects of the cost on the size of both sequences. We applied three methods of normalising the edit cost computation to this effect and observed a varying degree of performance on the datasets. The performance (F1-score) of the distance substitution kernel functions on the Recall, Refer and Therapy datasets is much closer. Normalization with the exponent of the number of common items performed best (F1-score) on the Clinical dataset. The variation of the normalising values generated with this kernel is more significant between pairs of sequences with similar items than with the other two normalising methods implemented. The pairs with a greater number of identical items are normalised by a large factor, thereby making their similarity score smaller and thus indicating they are much closer than without normalization. The results of our experiment indicate that this had a greater effect on the separability of the data.

The spectral ratio showed varying degrees of abstraction, implying substantial structural differences exist within the set of base kernels that were combined via MKL. This is a good indication that MKL achieves superior classification performance if the kernels to be combined are made up of a mixture of expressive as well as more general kernels. The variation seen is possibly due to the multiple kernels generated from variants of the edit distance measure. The underlying proximity measure differs, thus representing different structural abstractions such that a composite kernel has all that is necessary to find linear separability in the feature space. The empirical findings thus confirm the spectral ratio as a suitable evaluation method for assessing quality in this regard.

The success of deep learning in part depends on the availability of very large training datasets and computational resources. Deep learning constrained to a small sample-sized sparse bag-of-words

representation of EHR data consisting of 158 patients, as in the current case, is generally not feasible (techniques such as data augmentation or transfer learning can be applied to overcome this problem to a certain extent; however, such measures are beyond the scope of this study). Nonetheless, the results obtained from experiments show that the kernel framework presents an alternative strategy, in comparison to the performance obtained with bag-of-words features, for addressing classification tasks with uneven-length clinical sequences. Moreover, the computational efficiency of processing high-dimensional features with small sample-sized examples, as a feature of the kernel framework, constitutes a salient advantage over deep learning.

The results obtained from experimenting with the validation set show the kernel method had comparable, though less significant, performance results against those arising from the bag-of-words features. Applying MKL with the top 7 performing kernels achieved a comparable result to those of the single kernels; adding more kernels to the mixture degraded the performance (the same leave-one-out cross validation used on the primary dataset was adopted for the validation data experiment for comparative reasons). By validating our model on a dataset from a different domain, we are able to show the good classification performance obtained, which further supports our hypothesis that the proposed framework can be applied to uneven-length and irregularly sampled EHR data.

As a further note, while the LibSVM solver is in fact capable of handling non convex optimization problems, the proposed spectral modification to guarantee PSD kernels achieved a higher score than learning directly from indefinite kernels. Clipping and flipping the negative eigenvalues also performed better than shifting or squaring.

## 6.1 Related Work

In recent times, deep neural networks like convolutional neural networks (CNN) and recurrent neural networks (RNN) have been applied to address several healthcare analytical problems with significant results. This is primarily due to the near-human accuracy levels in various types of classification and prediction tasks, including images, text, speech, and video data in several domains [49]. The nature of EHR problems makes them amenable to applying deep learning methods. For instance, CNN is used in clinical diagnosis by classifying images such as MRI scans and X-rays. On the other hand, RNNs can capture the complex temporal dynamics in the longitudinal EHR data, thus making them the preferred architecture for several EHR modelling tasks, including sequential clinical event prediction, disease classification, and computational phenotyping [112, 260].

Despite their suitability and superior performance, deep learning methods still fall short of completely addressing the problems highlighted in the data. For instance, the representation of the raw EHR still requires a bit of engineering prior to input to the model. Time-based preprocessing as a way of addressing irregular sampling of the data was observed in the meta-analysis [217]. These methods include a form of embedding applied with post-padding with zeros to make the sequences equal in length. For instance, the study [29] truncated sequences longer than  $N$  and post-padded shorter sequences with zeros. Baytas et al. [28] proposed the time-aware LSTM (T-LSTM) as a method of addressing the temporal dynamics of longitudinal and irregularly sampled EHR. Gaussian process regression was applied to address the problem of irregular sampling in the study to estimate the temporal correlation between depression and suicide ideation [103]. Likewise in [143]. It can be seen that a more robust mechanism for achieving the goal is still desired.

Despite the excellent predictive performance, interpretability of deep learning methods still remains an obstacle [49, 130]. They are still regarded as “black box” models. There is also the issue of

overfitting [59, 244], since most of the experiments performed in the literature relied on a relatively small dataset, and computational cost [219]. The potential to overfit the data and yield inaccurate results stems from the requirement that deep learning methods perform better on large data sets. LSTM models, for instance, require lots of examples to outperform other methods [29, 204]. Dealing with heterogeneous entities remains an open problem for deep learning methods, as evidenced by the meta analysis of deep learning methods in healthcare [260].

This is in contrast to our proposed kernel framework that works well on small datasets, provides a principled way of addressing heterogeneous entities, and also, with elastic edit distance, overcomes the problems associated with irregularly sampled, arbitrary-length EHR data.

## 6.2 Future Work

The goal of this study was to see if the proposed kernel framework could be used to improve predictive modelling performance with EHR. Specifically, it sought to establish if the edit distance based kernel framework can be used to address the problem of arbitrary length and irregular sampling inherent in EHR data. A few issues were raised during the course of this research, which we propose to address in future research work:

- An initial goal was to test the proposed kernel framework with other kernel-based classifiers, such as the Gaussian process. This can be addressed by using the Gaussian process to model a time to an event. In continuation of the work done, we propose to model the duration from the first episode of elevated blood pressure to the actual date of confirmed onset of type 2 diabetes. This provides a means of predicting when a susceptible patient might develop the disease. In addition, any other case of interest with real-valued entities may also be undertaken.
- This work originally set out to model persistence with metformin in people with type 2 diabetes. The success in addressing problems with uneven-length and irregularly sampled EHR by retaining the data in its original form makes this a suitable analytic method. This offers a future research goal if “real-live” data is available.
- The distance substitution method applied a greedy search, with each data example serving as the zero-vector, to find the most suitable candidate. The results obtained from the experiments indicate a perfect candidate that yields a high level of predictive performance exists. This research did not explore the inherent characteristics and properties of such a perfect candidate. Therefore, it offers an open research problem that can be explored.

## 6.3 Limitations

Given the investigation’s scope, which is to establish the inherent suitability of featureless methods for EHR on the basis of their retention of all symbolic and real-valued data on an equal footing, it is not within the experimental scope (or part of the argument) to eliminate the inverse-corollary that feature-based methods are always information-losing. Indeed, there will invariably be many situations in which the intrinsic feature richness is such that this is not the case, and some overparameterized situations in which information loss (as opposed to noise loss) may be concretely useful.

## 6.4 Conclusion

The purpose of this study was to determine whether the proposed supervised machine learning kernel framework could be used in prediction modelling with uneven length heterogenous EHR data. We’ve

demonstrated the proposed edit distance-based kernel framework as a viable approach for overcoming the problems with symbolic EHR data, specifically in dealing with irregularly sampled, uneven-length longitudinal data. By using multi kernel learning, we also addressed the problem of heterogeneous entities by combining categorical data with real-valued numeric test measurements in the same model.

This research showed that bespoke edit distance variants, tailored to address specific problems, provide a means of incorporating domain knowledge. The elastic property of the distance addressed the uneven-length data problem, while the modular kernel construction process ensured standard methods of generating valid kernel functions.

Overall, the research findings show that the proposed framework has the potential to be implemented as a disease prognosis tool; thus providing a means to identify those at risk of developing type 2 diabetes from a prior incident of elevated blood pressure of 130/80 mm Hg at the primary care level. We believe implementing such a solution as part of the decision support system will help alleviate the burden of care in the long term management of type 2 diabetes. Furthermore, shifting to a healthier lifestyle is beneficial to both susceptible and healthy patients.

We propose that the outlined featureless edit kernel strategy may represent a generally preferable form of EHR-based machine learning on the basis of its implicit retention of all clinically relevant information that may otherwise be lost in the feature representation process.

## 6.5 Summary

In summary, this thesis answered the research question on the suitability of the edit distance-based kernel framework as a predictive modelling tool with problematic EHR data. There are many possibilities with implementing the proposed framework as a decision support plug-in tool within a GP IT system used in the context of routine primary care delivery. We hope that this thesis will provide empirical evidence to support this claim



# Appendix A

## Raw results

Kernel	F1	Acc	Sen	Spec	Mode	Trans	C	nSV	%-ve Eig
<b>Clinical</b>									
$k_{ed1}$	0.58	57.59 (49.58)	0.70	0.49	flip	None	0.0005	28.89 (0.43)	99.37
$k_{ed2}$	0.58	50.63 (50.15)	0.82	0.28	clip	Norm	0.25	8.99 (0.20)	94.94
$k_{ed3}$	0.58	55.06 (49.90)	0.73	0.42	square	Norm	0.25	120.61 (1.64)	56.96
$k_{ed4}$	0.61	56.33 (49.76)	0.80	0.39	nsd	None	128	57.78 (0.68)	49.37
$k_{ed5}$	0.53	40.51 (49.25)	0.80	0.12	nsd	Norm	0.25	4.00 (0.00)	99.37
$k_{ed8}$	0.56	43.04 (49.67)	0.86	0.12	nsd	Norm	0.25	4.00 (0.00)	77.22
<b>Recall</b>									
$k_{ed1}$	0.54	43.04 (49.67)	0.80	0.16	nsd	None	0.25	2.00 (0.00)	77.22
$k_{ed2}$	0.50	46.84 (50.06)	0.65	0.34	square	Norm	0.25	56.46 (1.09)	82.28
$k_{ed3}$	0.51	54.43 (49.96)	0.58	0.52	clip	Norm	128	15.82 (0.61)	59.49
$k_{ed4}$	0.51	55.06 (49.90)	0.56	0.54	clip	Norm	0.25	13.92 (0.33)	56.96
<b>Refer</b>									
$k_{ed1}$	0.59	42.41 (49.58)	0.98	0.02	clip	Norm	128	2.01 (0.08)	95.57
$k_{ed2}$	0.59	54.43 (49.96)	0.80	0.36	flip	None	0.25	35.92 (0.58)	91.77
$k_{ed3}$	0.58	48.10 (50.12)	0.85	0.22	clip	Norm	128	10.96 (0.33)	75.95
$k_{ed4}$	0.63	61.39 (48.84)	0.80	0.48	square	None	0.0005	123.20 (1.86)	63.29
<b>Repeat</b>									
$k_{ed1}$	0.60	43.67 (49.76)	1.00	0.03	clip	None	0.25	2.00 (0.00)	98.73
$k_{ed2}$	0.67	65.82 (47.58)	0.82	0.54	square	None	0.25	56.14 (0.94)	93.67
$k_{ed3}$	0.56	51.27 (50.14)	0.74	0.35	flip	Norm	0.25	48.11 (0.86)	63.92
$k_{ed4}$	0.57	54.43 (49.96)	0.71	0.42	square	None	0.0005	102.03 (1.68)	48.73
<b>Test</b>									
$k_{ed1}$	0.67	63.29 (48.35)	0.88	0.46	clip	None	0.0005	36.00 (0.00)	81.01
$k_{ed2}$	0.52	50.63 (50.15)	0.65	0.40	flip	Norm	128	40.90 (0.63)	85.44
$k_{ed3}$	0.58	43.04 (49.67)	0.94	0.07	clip	None	0.25	6.00 (0.23)	60.76
$k_{ed4}$	0.57	56.96 (49.67)	0.67	0.50	flip	Norm	0.25	31.68 (0.91)	50.00
<b>Therapy</b>									
$k_{ed1}$	0.59	41.77 (49.48)	0.98	0.01	nsd	None	0.25	2.00 (0.00)	99.37
$k_{ed2}$	0.60	55.70 (49.83)	0.79	0.39	square	None	0.25	96.39 (1.71)	93.04
$k_{ed3}$	0.59	48.73 (50.14)	0.89	0.20	nsd	None	0.25	2.00 (0.00)	50.00
$k_{ed4}$	0.53	55.06 (49.90)	0.61	0.51	square	Norm	0.25	130.80 (0.99)	46.84
$k_{ed10}$	0.59	42.41 (49.58)	1.00	0.01	nsd	None	0.25	2.00 (0.00)	99.37
<b>All data</b>									
$k_{ed1}$	0.60	55.70 (49.83)	0.79	0.39	flip	None	0.0005	31.77 (0.49)	99.37
$k_{ed2}$	0.58	52.53 (50.09)	0.80	0.33	flip	Norm	0.25	97.72 (0.69)	94.94

Pseudo Kernel

Table A.1 –

<b>K</b>	<b>F1</b>	<b>Acc</b>	<b>Sen</b>	<b>Spec</b>	<b>Mode</b>	<b>Trans</b>	<b>C</b>	<b>nSV</b>	<b>%-ve Eig</b>
$k_{ed3}$	0.60	51.90 (50.12)	0.85	0.28	clip	Norm	0.25	18.06 (0.54)	58.86
$k_{ed4}$	0.60	48.73 (50.14)	0.92	0.17	square	Norm	0.25	47.26 (0.89)	50.00
$k_{ed9}$	0.60	51.27 (50.14)	0.88	0.25	square	Norm	0.25	75.28 (1.00)	50.63

Table A.1: Table showing classification performance achieved with the pseudo kernels executed on various datasets

<b>Kernel</b>	<b>F1</b>	<b>Acc</b>	<b>Sen</b>	<b>Spec</b>	<b>Mode</b>	<b>Trans</b>	<b>C</b>	<b>nSV</b>	<b>%-ve Eig</b>
<b>Clinical</b>									
$k_{s\_ed1}$	0.00	58.23 (49.48)	0.00	1.00	psd	None	9.54E-07	131.16 (0.99)	0.00
$k_{s\_ed2}$	0.61	58.86 (49.37)	0.76	0.47	psd	None	0.25	115.37 (0.50)	0.00
$k_{s\_ed3}$	0.62	58.86 (49.37)	0.83	0.41	nsd	None	0.25	35.03 (0.52)	1.90
$k_{s\_ed4}$	0.60	54.43 (49.96)	0.83	0.34	square	Norm	128	29.93 (0.38)	49.37
$k_{s\_ed5}$	0.44	58.86 (49.37)	0.39	0.73	psd	Norm	0.25	8.97 (0.26)	0.00
$k_{s\_ed6}$	0.60	44.30 (49.83)	1.00	0.04	clip	None	128	2.98 (0.14)	98.73
$k_{s\_ed7}$	0.61	51.27 (50.14)	0.91	0.23	psd	Norm	128	43.37 (1.17)	0.00
$k_{s\_ed8}$	0.46	57.59 (49.58)	0.44	0.67	clip	Norm	0.25	9.93 (0.28)	17.72
<b>Recall</b>									
$k_{s\_ed1}$	0.29	59.49 (49.25)	0.20	0.88	nsd	None	128	106.36 (0.60)	6.96
$k_{s\_ed2}$	0.61	48.73 (50.14)	0.94	0.16	nsd	None	0.25	44.85 (0.76)	10.13
$k_{s\_ed3}$	0.51	55.06 (49.90)	0.56	0.54	nsd	None	128	107.87 (0.66)	6.96
$k_{s\_ed4}$	0.57	50.00 (50.16)	0.79	0.29	nsd	None	0.25	78.59 (0.63)	7.59
<b>Refer</b>									
$k_{s\_ed1}$	0.56	51.27 (50.14)	0.74	0.35	nsd	None	128	84.47 (0.97)	2.53
$k_{s\_ed2}$	0.64	60.76 (48.98)	0.82	0.46	flip	Norm	0.25	74.64 (0.82)	4.43
$k_{s\_ed3}$	0.56	48.10 (50.12)	0.80	0.25	nsd	None	128	87.51 (0.65)	3.16
$k_{s\_ed4}$	0.62	58.23 (49.48)	0.82	0.41	nsd	None	128	65.56 (0.64)	2.53
<b>Repeat</b>									
$k_{s\_ed1}$	0.38	55.06 (49.90)	0.33	0.71	psd	None	128	126.94 (0.66)	0.00
$k_{s\_ed2}$	0.60	58.23 (49.48)	0.76	0.46	flip	None	0.25	102.44 (0.54)	2.53
$k_{s\_ed3}$	0.53	53.80 (50.01)	0.62	0.48	psd	None	128	129.79 (0.91)	0.00
$k_{s\_ed4}$	0.62	59.49 (49.25)	0.79	0.46	flip	Norm	128	106.33 (0.64)	3.16
<b>Test</b>									
$k_{s\_ed1}$	0.24	43.67 (49.76)	0.21	0.60	nsd	None	0.25	128.99 (1.46)	1.90
$k_{s\_ed2}$	0.47	55.06 (49.90)	0.48	0.60	shift	None	9.54E-07	60.73 (1.82)	6.33
$k_{s\_ed3}$	0.55	53.80 (50.01)	0.68	0.43	nsd	None	128	73.75 (0.71)	3.16
$k_{s\_ed4}$	0.60	55.70 (49.83)	0.80	0.38	nsd	None	0.25	31.97 (0.42)	50.00
<b>Therapy</b>									
$k_{s\_ed1}$	0.00	58.23 (49.48)	0.00	1.00	psd	None	9.54E-07	131.16 (0.99)	0.00
$k_{s\_ed2}$	0.51	62.03 (48.69)	0.48	0.72	flip	None	0.25	108.53 (0.53)	3.16
$k_{s\_ed3}$	0.09	58.23 (49.48)	0.05	0.97	psd	None	128	156.06 (0.24)	0.00
$k_{s\_ed4}$	0.51	56.33 (49.76)	0.55	0.58	nsd	None	128	101.67 (0.67)	18.99
$k_{s\_ed10}$	0.00	58.23 (49.48)	0.00	1.00	psd	None	9.54E-07	131.16 (0.99)	0.00
<b>All data</b>									
$k_{s\_ed1}$	0.00	58.23 (49.48)	0.00	1.00	psd	None	9.54E-07	131.16 (0.99)	0.00
$k_{s\_ed2}$	0.56	55.06 (49.90)	0.70	0.45	flip	None	0.25	96.30 (0.86)	2.53

Edit Similarity Kernel

Table A.2 –

<b>K</b>	<b>F1</b>	<b>Acc</b>	<b>Sen</b>	<b>Spec</b>	<b>Mode</b>	<b>Trans</b>	<b>C</b>	<b>nSV</b>	<b>%-ve Eig</b>
$k_{s\_ed3}$	0.54	53.16 (50.06)	0.64	0.46	psd	None	128	128.94 (0.75)	0.00
$k_{s\_ed4}$	0.61	53.16 (50.06)	0.85	0.30	square	Norm	0.25	36.87 (0.48)	50.63
$k_{s\_ed9}$	0.61	60.13 (49.12)	0.73	0.51	clip	Norm	128	57.91 (0.56)	39.87

Table A.2: Table showing classification performance achieved with the edit similarity kernels executed on multiple datasets

<b>Kernel</b>	<b>F1</b>	<b>Acc</b>	<b>Sen</b>	<b>Spec</b>	<b>Mode</b>	<b>Trans</b>	<b>C</b>	<b>nSV</b>	<b>%-ve Eig</b>
<b>Clinical</b>									
$k_{G\_ed1}$	0.33	60.76 (48.98)	0.23	0.88	psd	Raw	0.25	18.89 (0.32)	0.00
$k_{G\_ed2}$	0.63	55.70 (49.83)	0.91	0.30	clip	Raw	0.25	16.94 (0.42)	4.43
$k_{G\_ed3}$	0.60	50.63 (50.15)	0.88	0.24	flip	Norm	0.25	23.15 (0.63)	35.44
$k_{G\_ed4}$	0.62	58.23 (49.48)	0.83	0.40	square	Norm	128	30.20 (0.78)	49.37
$k_{G\_ed5}$	0.59	48.10 (50.12)	0.91	0.17	clip	Norm	0.25	22.94 (0.33)	48.73
$k_{G\_ed6}$	0.63	52.53 (50.09)	0.95	0.22	square	Norm	0.25	13.89 (0.48)	50.00
$k_{G\_ed7}$	0.63	58.86 (49.37)	0.86	0.39	clip	Norm	128	70.96 (0.56)	5.06
$k_{G\_ed8}$	0.61	49.37 (50.15)	0.95	0.16	clip	Norm	0.25	18.85 (0.53)	49.37
<b>Recall</b>									
$k_{G\_ed1}$	0.57	65.82 (47.58)	0.55	0.74	shift	Raw	9.54E-07	80.70 (4.60)	13.29
$k_{G\_ed2}$	0.62	51.27 (50.14)	0.95	0.20	clip	Norm	0.25	37.97 (0.71)	12.03
$k_{G\_ed3}$	0.53	60.13 (49.12)	0.55	0.64	shift	Raw	9.54E-07	77.16 (3.17)	39.24
$k_{G\_ed4}$	0.55	54.43 (49.96)	0.65	0.47	raw	Raw	128	109.18 (1.05)	6.96
<b>Refer</b>									
$k_{G\_ed1}$	0.59	41.77 (49.48)	0.97	0.02	raw	Raw	0.25	15.13 (0.69)	2.53
$k_{G\_ed2}$	0.64	62.66 (48.52)	0.80	0.50	raw	Raw	128	16.94 (0.33)	7.59
$k_{G\_ed3}$	0.62	58.23 (49.48)	0.80	0.42	clip	Raw	0.25	17.04 (0.41)	22.15
$k_{G\_ed4}$	0.64	60.13 (49.12)	0.85	0.42	clip	Raw	128	11.89 (0.64)	36.71
<b>Repeat</b>									
$k_{G\_ed1}$	0.59	47.47 (50.09)	0.91	0.16	psd	Raw	128	42.68 (0.57)	0.00
$k_{G\_ed2}$	0.66	65.82 (47.58)	0.77	0.58	raw	Raw	128	22.80 (0.64)	5.06
$k_{G\_ed3}$	0.60	46.84 (50.06)	0.92	0.14	flip	Raw	0.25	24.91 (0.46)	34.18
$k_{G\_ed4}$	0.62	56.96 (49.67)	0.85	0.37	square	Raw	0.25	33.03 (0.44)	48.10
<b>Test</b>									
$k_{G\_ed1}$	0.52	49.37 (50.15)	0.67	0.37	raw	Raw	0.25	78.58 (0.64)	6.33
$k_{G\_ed2}$	0.45	46.20 (50.01)	0.52	0.42	clip	Norm	128	18.13 (0.62)	12.03
$k_{G\_ed3}$	0.58	58.86 (49.37)	0.68	0.52	raw	Raw	128	10.03 (0.33)	31.01
$k_{G\_ed4}$	0.63	56.33 (49.76)	0.89	0.33	square	Norm	0.25	9.01 (0.14)	50.00
<b>Therapy</b>									
$k_{G\_ed1}$	0.61	48.73 (50.14)	0.94	0.16	psd	Raw	128	16.92 (0.28)	0.00
$k_{G\_ed2}$	0.62	60.76 (48.98)	0.76	0.50	clip	Norm	128	51.44 (1.98)	5.70
$k_{G\_ed3}$	0.54	56.33 (49.76)	0.62	0.52	square	Norm	0.25	40.16 (0.81)	48.73
$k_{G\_ed4}$	0.56	51.90 (50.12)	0.74	0.36	flip	Norm	0.25	77.85 (0.72)	42.41
$k_{G\_ed10}$	0.58	56.33 (49.76)	0.73	0.45	psd	Raw	128	11.02 (0.29)	0.00
<b>All data</b>									
$k_{G\_ed1}$	0.44	53.80 (50.01)	0.44	0.61	psd	Raw	128	77.84 (0.37)	0.00
$k_{G\_ed2}$	0.62	52.53 (50.09)	0.92	0.24	square	Raw	0.25	15.99 (0.40)	4.43

Gaussian Edit Kernel

Table A.3 –

<b>K</b>	<b>F1</b>	<b>Acc</b>	<b>Sen</b>	<b>Spec</b>	<b>Mode</b>	<b>Trans</b>	<b>C</b>	<b>nSV</b>	<b>%-ve Eig</b>
$k_{G\_ed3}$	0.61	53.16 (50.06)	0.85	0.30	flip	Raw	0.25	22.94 (0.40)	40.51
$k_{G\_ed4}$	0.64	54.43 (49.96)	0.94	0.26	square	Raw	0.25	33.96 (1.17)	50.00
$k_{G\_ed9}$	0.63	61.39 (48.84)	0.77	0.50	clip	Raw	128	48.69 (0.71)	42.41

Table A.3: Table showing classification performance achieved with the Gaussian edit kernels executed on multiple datasets

<b>Kernel</b>	<b>F1</b>	<b>Acc</b>	<b>Sen</b>	<b>Spec</b>	<b>Mode</b>	<b>Trans</b>	<b>C</b>	<b>nSV</b>	<b>%-ve Eig</b>
<b>Clinical</b>									
$k_{r\_ed1}$	0.00	58.23 (49.48)	0.00	1.00	psd	None	9.54E-07	131.16 (0.99)	0.00
$k_{r\_ed2}$	0.62	58.86 (49.37)	0.83	0.41	nsd	None	0.25	35.03 (0.52)	1.90
$k_{r\_ed3}$	0.61	58.86 (49.37)	0.76	0.47	psd	None	0.25	115.37 (0.50)	0.00
$k_{r\_ed4}$	0.61	48.73 (50.14)	0.95	0.15	flip	Norm	128	27.89 (0.52)	50.00
$k_{r\_ed5}$	0.59	44.94 (49.90)	0.94	0.10	clip	Norm	0.25	8.01 (0.20)	10.76
$k_{r\_ed6}$	0.61	55.06 (49.90)	0.83	0.35	nsd	Norm	0.25	8.00 (0.11)	24.05
<b>Recall</b>									
$k_{r\_ed1}$	0.33	60.13 (49.12)	0.23	0.87	nsd	None	128	100.64 (0.68)	6.96
$k_{r\_ed2}$	0.62	51.27 (50.14)	0.95	0.20	nsd	None	0.25	41.74 (0.86)	10.76
$k_{r\_ed3}$	0.51	52.53 (50.09)	0.59	0.48	nsd	None	0.25	100.87 (0.67)	7.59
$k_{r\_ed4}$	0.57	50.00 (50.16)	0.79	0.29	nsd	None	0.25	78.59 (0.63)	8.23
<b>Refer</b>									
$k_{r\_ed1}$	0.57	51.27 (50.14)	0.77	0.33	nsd	None	0.25	79.72 (0.48)	1.90
$k_{r\_ed2}$	0.64	60.76 (48.98)	0.82	0.46	flip	Norm	0.25	74.64 (0.82)	4.43
$k_{r\_ed3}$	0.57	46.20 (50.01)	0.83	0.20	nsd	None	128	79.04 (0.53)	3.16
$k_{r\_ed4}$	0.63	59.49 (49.25)	0.82	0.43	nsd	None	0.25	59.98 (0.81)	3.16
<b>Repeat</b>									
$k_{r\_ed1}$	0.45	57.59 (49.58)	0.42	0.68	nsd	None	128	121.29 (0.50)	0.63
$k_{r\_ed2}$	0.60	58.23 (49.48)	0.76	0.46	flip	None	0.25	102.44 (0.54)	2.53
$k_{r\_ed3}$	0.55	53.80 (50.01)	0.68	0.43	psd	None	128	123.13 (0.77)	0.00
$k_{r\_ed4}$	0.62	59.49 (49.25)	0.79	0.46	flip	Norm	128	106.33 (0.64)	2.53
<b>Test</b>									
$k_{r\_ed1}$	0.31	44.30 (49.83)	0.30	0.54	nsd	None	0.25	127.61 (1.28)	3.80
$k_{r\_ed2}$	0.50	58.23 (49.48)	0.50	0.64	shift	None	9.54E-07	52.27 (1.78)	5.06
$k_{r\_ed3}$	0.56	54.43 (49.96)	0.68	0.45	nsd	None	128	65.64 (0.72)	3.80
$k_{r\_ed4}$	0.60	55.70 (49.83)	0.80	0.38	nsd	None	0.25	31.97 (0.42)	50.00
<b>Therapy</b>									
$k_{r\_ed1}$	0.00	58.23 (49.48)	0.00	1.00	psd	None	9.54E-07	131.16 (0.99)	0.00
$k_{r\_ed2}$	0.52	58.86 (49.37)	0.53	0.63	flip	None	128	109.22 (0.87)	3.16
$k_{r\_ed3}$	0.14	59.49 (49.25)	0.08	0.97	psd	None	128	154.87 (0.38)	0.00
$k_{r\_ed4}$	0.52	56.96 (49.67)	0.55	0.59	nsd	None	128	96.38 (0.74)	19.62
$k_{r\_ed10}$	0.57	48.10 (50.12)	0.82	0.24	psd	None	128	17.90 (0.30)	0.00
<b>All data</b>									
$k_{r\_ed1}$	0.00	58.23 (49.48)	0.00	1.00	psd	None	9.54E-07	131.16 (0.99)	0.00
$k_{r\_ed2}$	0.57	53.80 (50.01)	0.71	0.41	flip	None	0.25	94.54 (0.78)	2.53
$k_{r\_ed3}$	0.54	53.16 (50.06)	0.64	0.46	psd	None	128	128.94 (0.75)	0.00
$k_{r\_ed4}$	0.63	52.53 (50.09)	0.94	0.23	nsd	None	128	26.77 (0.58)	50.00

Rational Quadratic Edit Kernel

Table A.4 –

<b>K</b>	<b>F1</b>	<b>Acc</b>	<b>Sen</b>	<b>Spec</b>	<b>Mode</b>	<b>Trans</b>	<b>C</b>	<b>nSV</b>	<b>%-ve Eig</b>
$k_{r\_ed9}$	0.63	60.76 (48.98)	0.79	0.48	clip	None	128	57.79 (0.61)	39.24

Table A.4: able showing classification performance achieved with the Rational Quadratic edit kernels executed on multiple datasets

<b>Kernel</b>	<b>F1</b>	<b>Acc</b>	<b>Sen</b>	<b>Spec</b>	<b>Mode</b>	<b>Trans</b>	<b>C</b>	<b>nSV</b>	<b>%-ve Eig</b>
<b>Clinical</b>									
$k_{p\_ed1}$	0.00	58.23 (49.48)	0.00	1.00	psd	None	9.54E-07	131.16 (0.99)	0.00
$k_{p\_ed2}$	0.63	60.76 (48.98)	0.79	0.48	nsd	None	0.25	41.82 (0.46)	1.90
$k_{p\_ed3}$	0.61	62.66 (48.52)	0.70	0.58	psd	None	0.25	145.03 (0.67)	0.00
$k_{p\_ed4}$	0.62	63.29 (48.35)	0.71	0.58	square	Norm	0.25	136.99 (1.67)	48.10
$k_{p\_ed5}$	0.59	41.77 (49.48)	1.00	0.00	psd	None	0.25	4.00 (0.00)	0.00
$k_{p\_ed6}$	0.61	55.06 (49.90)	0.82	0.36	square	None	2.50E-01	4.02 (0.14)	98.73
$k_{p\_ed7}$	0.60	55.06 (49.90)	0.79	0.38	psd	None	0.25	19.74 (0.86)	0.00
$k_{p\_ed8}$	0.59	41.77 (49.48)	1.00	0.00	nsd	None	0.25	4.00 (0.00)	17.09
<b>Recall</b>									
$k_{p\_ed1}$	0.28	60.76 (48.98)	0.18	0.91	nsd	None	0.25	114.77 (0.81)	8.23
$k_{p\_ed2}$	0.61	48.73 (50.14)	0.94	0.16	flip	None	0.25	50.40 (1.18)	10.13
$k_{p\_ed3}$	0.50	52.53 (50.09)	0.56	0.50	nsd	None	0.25	117.64 (0.54)	8.86
$k_{p\_ed4}$	0.56	55.06 (49.90)	0.70	0.45	nsd	None	0.25	90.08 (0.81)	6.96
<b>Refer</b>									
$k_{p\_ed1}$	0.54	53.16 (50.06)	0.65	0.45	nsd	None	0.25	97.42 (0.49)	3.16
$k_{p\_ed2}$	0.64	61.39 (48.84)	0.82	0.47	clip	Norm	0.25	63.78 (0.79)	5.70
$k_{p\_ed3}$	0.53	48.73 (50.14)	0.68	0.35	nsd	None	0.25	97.43 (0.58)	1.90
$k_{p\_ed4}$	0.63	60.76 (48.98)	0.80	0.47	nsd	None	0.25	91.41 (0.70)	3.16
<b>Repeat</b>									
$k_{p\_ed1}$	0.34	56.96 (49.67)	0.27	0.78	psd	None	0.25	138.61 (0.49)	0.00
$k_{p\_ed2}$	0.63	66.46 (47.36)	0.70	0.64	flip	None	0.25	100.70 (0.65)	2.53
$k_{p\_ed3}$	0.52	57.59 (49.58)	0.56	0.59	psd	None	0.25	140.53 (0.77)	0.00
$k_{p\_ed4}$	0.61	64.56 (47.99)	0.67	0.63	nsd	None	0.25	135.80 (0.75)	0.63
<b>Test</b>									
$k_{p\_ed1}$	0.00	58.23 (49.48)	0.00	1.00	nsd	None	9.54E-07	131.16 (0.99)	3.80
$k_{p\_ed2}$	0.52	56.96 (49.67)	0.55	0.59	shift	None	9.54E-07	79.04 (3.33)	4.43
$k_{p\_ed3}$	0.55	53.16 (50.06)	0.67	0.43	nsd	None	0.25	125.18 (0.93)	4.43
$k_{p\_ed4}$	0.60	58.86 (49.37)	0.74	0.48	clip	Norm	128	36.49 (0.78)	46.84
<b>Therapy</b>									
$k_{p\_ed1}$	0.00	58.23 (49.48)	0.00	1.00	psd	None	9.54E-07	131.16 (0.99)	0.00
$k_{p\_ed2}$	0.53	65.19 (47.79)	0.47	0.78	flip	None	0.25	110.59 (0.49)	1.90
$k_{p\_ed3}$	0.00	58.23 (49.48)	0.00	1.00	psd	None	9.54E-07	131.16 (0.99)	0.00
$k_{p\_ed4}$	0.53	41.77 (49.48)	0.79	0.15	square	Norm	2.50E-01	70.21 (2.06)	16.46
$k_{p\_ed10}$	0.00	58.23 (49.48)	0.00	1.00	psd	None	9.54E-07	131.16 (0.99)	0.00
<b>All data</b>									
$k_{p\_ed1}$	0.00	58.23 (49.48)	0.00	1.00	psd	None	9.54E-07	131.16 (0.99)	0.00
$k_{p\_ed2}$	0.59	56.96 (49.67)	0.73	0.46	flip	None	0.25	100.66 (0.52)	1.90
$k_{p\_ed3}$	0.49	54.43 (49.96)	0.53	0.55	psd	None	0.25	139.89 (0.56)	0.00
$k_{p\_ed4}$	0.61	51.90 (50.12)	0.91	0.24	nsd	Norm	0.25	12.06 (0.31)	50.00

Polynomial Edit Kernel

Table A.5 –

K	F1	Acc	Sen	Spec	Mode	Trans	C	nSV	%-ve Eig
$k_{p\_ed9}$	0.63	61.39 (48.84)	0.79	0.49	clip	None	2.50E-01	57.70 (0.58)	39.24

Table A.5: Table showing classification performance achieved with the polynomial edit kernels executed on multiple datasets

Kernel	F1	Acc	Sen	Spec	Mode	Trans	C	nSV	%-ve Eig
<b>Clinical</b>									
$k_{ds\_ed1}$	0.64	58.60 (49.41)	0.88	0.38	psd	Raw	0.25	17 (0)	0.00
$k_{ds\_ed2}$	0.63	58.60 (49.41)	0.86	0.39	psd	Raw	128	17 (0)	0.00
$k_{ds\_ed3}$	0.72	70.70 (45.66)	0.91	0.57	clip	Raw	0.00048828125	90 (0)	29.30
$k_{ds\_ed4}$	0.74	75.80 (42.97)	0.80	0.73	flip	Norm	0.25	40 (1)	22.29
<b>Recall</b>									
$k_{ds\_ed1}$	0.63	52.23 (50.11)	0.95	0.21	flip	Norm	0.25	36 (1)	24.20
$k_{ds\_ed2}$	0.62	50.32 (50.16)	0.95	0.18	clip	Norm	0.25	43 (1)	12.10
$k_{ds\_ed3}$	0.63	52.23 (50.11)	0.95	0.21	flip	Norm	0.25	36 (1)	24.20
$k_{ds\_ed4}$	0.63	52.23 (50.11)	0.95	0.21	flip	Norm	0.25	34 (1)	24.84
<b>Refer</b>									
$k_{ds\_ed1}$	0.65	61.15 (48.90)	0.80	0.47	flip	Norm	0.25	83 (1)	3.18
$k_{ds\_ed2}$	0.65	61.15 (48.90)	0.85	0.45	clip	Raw	128	52 (1)	2.55
$k_{ds\_ed3}$	0.65	61.15 (48.90)	0.80	0.47	flip	Norm	0.25	83 (1)	3.18
$k_{ds\_ed4}$	0.65	61.15 (48.90)	0.80	0.47	flip	Norm	0.25	83 (1)	3.18
<b>Repeat</b>									
$k_{ds\_ed1}$	0.69	69.43 (46.22)	0.80	0.62	clip	Raw	0.0005	93 (1)	0.64
$k_{ds\_ed2}$	0.64	66.88 (47.22)	0.71	0.64	clip	Norm	0.25	130 (0)	0.64
$k_{ds\_ed3}$	0.69	69.43 (46.22)	0.82	0.60	clip	Raw	0.0005	90 (1)	3.82
$k_{ds\_ed4}$	0.66	67.52 (46.98)	0.76	0.62	clip	Norm	0.25	114 (1)	0.64
<b>Test</b>									
$k_{ds\_ed1}$	0.64	64.97 (47.86)	0.76	0.57	flip	Norm	0.25	37 (1)	16.56
$k_{ds\_ed2}$	0.66	66.88 (47.22)	0.76	0.60	flip	Norm	0.25	49 (1)	5.10
$k_{ds\_ed3}$	0.62	59.87 (49.17)	0.80	0.45	clip	Raw	0.25	17 (0)	25.48
$k_{ds\_ed4}$	0.68	63.06 (48.42)	0.95	0.40	clip	Raw	128	102 (0)	49.04
<b>Therapy</b>									
$k_{ds\_ed1}$	0.67	66.88 (47.22)	0.80	0.58	psd	norm	0.25	89 (1)	0.00
$k_{ds\_ed2}$	0.67	65.61 (47.65)	0.82	0.54	psd	Raw	0.25	128 (1)	0.00
$k_{ds\_ed3}$	0.67	66.24 (47.44)	0.80	0.56	flip	Norm	128	55 (1)	7.01
$k_{ds\_ed4}$	0.67	68.15 (46.74)	0.77	0.62	clip	Norm	0.25	80 (1)	1.27
<b>All data</b>									
$k_{ds\_ed1}$	0.64	57.32 (49.62)	0.89	0.34	psd	Norm'	128	30 (1)	0.00
$k_{ds\_ed2}$	0.64	54.14 (49.99)	0.94	0.25	psd	Raw	0.25	57 (1)	0.00
$k_{ds\_ed3}$	0.73	75.16 (43.35)	0.80	0.71	flip	Raw	0.0005	110 (1)	31.21
$k_{ds\_ed4}$	0.74	70.70 (45.66)	0.98	0.51	square	Norm	0.0005	120 (1)	21.02

Table A.6: Table showing classification performance achieved with the edit distance substitution kernels executed on various datasets

Kernel	F1	Acc	Sen	Spec	Mode	Trans	C	nSV	%-ve Eig
--------	----	-----	-----	------	------	-------	---	-----	----------

Table A.7: Table showing classification performance achieved with the edit template matching kernels executed on various datasets

Size	Number	Kernel	F1	Acc	Sen	Spec	C	nSV
158	12	$k_{G\_ed1}$	0.81	85.44 (35.38)	0.73	0.95	3.25E-05	69 (11)
		$k_{G\_ed2}$	0.04	3.16 (17.56)	0.05	0.02	2.00E+00	9 (5)
		$k_{G\_ed3}$	0.70	65.19 (47.79)	0.98	0.41	7.05E-03	116 (8)
		$k_{G\_ed4}$	0.82	83.54 (37.20)	0.91	0.78	2.00E+00	2 (1)
134	12	$k_{G\_ed1}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	2 (0)
		$k_{G\_ed2}$	0.62	44.78 (49.91)	1.00	0.00	7.05E-03	119 (6)
		$k_{G\_ed3}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	2 (0)
		$k_{G\_ed4}$	0.63	47.76 (50.14)	0.98	0.07	2.00E+00	3 (2)
115	12	$k_{G\_ed1}$	0.65	50.43 (50.22)	1.00	0.08	3.25E-05	53 (4)
		$k_{G\_ed2}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	12 (2)
		$k_{G\_ed3}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	2 (0)
		$k_{G\_ed4}$	0.65	50.43 (50.22)	1.00	0.08	7.05E-03	107 (4)
104	12	$k_{G\_ed1}$	0.64	47.12 (50.16)	1.00	0.00	2.00E+00	2 (0)
		$k_{G\_ed2}$	0.64	47.12 (50.16)	1.00	0.00	7.05E-03	94 (3)
		$k_{G\_ed3}$	0.64	47.12 (50.16)	1.00	0.00	2.00E+00	2 (0)
		$k_{G\_ed4}$	0.64	48.08 (50.20)	0.98	0.04	2.00E+00	10 (2)

Table A.8: MKL classification performance achieved by combining 12 edit kernels on Clinical Table

Size	Number	Kernel	F1	Acc	Sen	Spec	C	nSV
158	12	$k_{G\_ed1}$	0.74	71.52 (45.28)	0.97	0.53	7.05E-03	136 (6)
		$k_{G\_ed2}$	NaN	58.23 (49.48)	0.00	1.00	7.05E-03	137 (1)
		$k_{G\_ed3}$	0.70	67.09 (47.14)	0.91	0.50	7.05E-03	125 (9)
		$k_{G\_ed4}$	0.67	68.35 (46.66)	0.76	0.63	7.05E-03	111 (13)
134	12	$k_{G\_ed1}$	0.62	45.52 (49.99)	1.00	0.01	2.00E+00	3 (0)
		$k_{G\_ed2}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	105 (7)
		$k_{G\_ed3}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	2 (0)
		$k_{G\_ed4}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	1 (1)
115	12	$k_{G\_ed1}$	0.63	46.96 (50.13)	1.00	0.02	2.00E+00	3 (0)
		$k_{G\_ed2}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	93 (7)
		$k_{G\_ed3}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	2 (0)
		$k_{G\_ed4}$	0.63	46.96 (50.13)	1.00	0.02	2.00E+00	3 (0)

Recall Table

Table A.9 –

Size	Number	Kernel	F1	Acc	Sen	Spec	C	nSV
104	12	$k_{G\_ed1}$	0.65	48.08 (50.20)	1.00	0.02	2.00E+00	3 (0)
		$k_{G\_ed2}$	0.64	47.12 (50.16)	1.00	0.00	2.00E+00	88 (7)
		$k_{G\_ed3}$	0.64	47.12 (50.16)	1.00	0.00	2.00E+00	2 (0)
		$k_{G\_ed4}$	0.64	47.12 (50.16)	1.00	0.00	2.00E+00	4 (0)

Table A.9: MKL classification performance achieved by combining 12 edit kernels on Recall Table

Size	Number	Kernel	F1	Acc	Sen	Spec	C	nSV
158	12	$k_{G\_ed1}$	0.56	39.87 (49.12)	0.92	0.02	7.05E-03	128 (10)
		$k_{G\_ed2}$	NaN	0.63 (7.96)	0.00	1.00	7.05E-03	2 (11)
		$k_{G\_ed3}$	0.49	33.54 (47.36)	0.79	0.01	7.05E-03	127 (13)
		$k_{G\_ed4}$	0.66	55.70 (49.83)	1.00	0.24	7.05E-03	133 (16)
134	12	$k_{G\_ed1}$	0.74	69.40 (46.25)	0.97	0.47	7.05E-03	105 (7)
		$k_{G\_ed2}$	NaN	0.75 (8.64)	0.00	1.00	1.05E-04	2 (11)
		$k_{G\_ed3}$	0.70	65.67 (47.66)	0.88	0.47	7.05E-03	104 (7)
		$k_{G\_ed4}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	3 (0)
115	12	$k_{G\_ed1}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	2 (1)
		$k_{G\_ed2}$	NaN	0.87 (9.33)	0.00	1.00	1.05E-04	2 (10)
		$k_{G\_ed3}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	2 (1)
		$k_{G\_ed4}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	5 (0)
104	12	$k_{G\_ed1}$	0.72	63.46 (48.39)	1.00	0.31	7.05E-03	88 (3)
		$k_{G\_ed2}$	NaN	0.96 (9.81)	0.00	1.00	1.05E-04	2 (10)
		$k_{G\_ed3}$	0.76	70.19 (45.96)	1.00	0.44	7.05E-03	83 (5)
		$k_{G\_ed4}$	0.67	52.88 (50.16)	1.00	0.11	7.05E-03	91 (4)

Table A.10: MKL classification performance achieved by combining 12 edit kernels on Refer Table

Size	Number	Kernel	F1	Acc	Sen	Spec	C	nSV
158	12	$k_{G\_ed1}$	0.99	99.37 (7.96)	0.98	1.00	7.05E-03	153 (2)
		$k_{G\_ed2}$	0.05	4.43 (20.64)	0.06	0.03	2.00E+00	11 (2)
		$k_{G\_ed3}$	0.80	79.75 (40.32)	0.98	0.66	7.05E-03	136 (4)
		$k_{G\_ed4}$	0.67	58.86 (49.37)	1.00	0.29	7.05E-03	121 (5)
134	12	$k_{G\_ed1}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	2 (0)
		$k_{G\_ed2}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	15 (3)
		$k_{G\_ed3}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	2 (0)
		$k_{G\_ed4}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	2 (0)

Repeat Table



Table A.11 –

Size	Number	Kernel	F1	Acc	Sen	Spec	C	nSV
115	12	$k_{G\_ed1}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	1 (0)
		$k_{G\_ed2}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	22 (3)
		$k_{G\_ed3}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	1 (0)
		$k_{G\_ed4}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	1 (1)
104	12	$k_{G\_ed1}$	0.64	47.12 (50.16)	1.00	0.00	2.00E+00	1 (0)
		$k_{G\_ed2}$	0.64	47.12 (50.16)	1.00	0.00	2.00E+00	19 (2)
		$k_{G\_ed3}$	0.64	47.12 (50.16)	1.00	0.00	2.00E+00	1 (0)
		$k_{G\_ed4}$	0.64	47.12 (50.16)	1.00	0.00	2.00E+00	1 (1)

Table A.11: MKL classification performance achieved by combining 12 edit kernels on Repeat Table

Size	Number	Kernel	F1	Acc	Sen	Spec	C	nSV
158	12	$k_{G\_ed1}$	0.99	99.37 (7.96)	0.98	1.00	7.05E-03	113 (6)
		$k_{G\_ed2}$	NaN	58.23 (49.48)	0.00	1.00	7.05E-03	140 (1)
		$k_{G\_ed3}$	0.98	98.10 (13.69)	1.00	0.97	2.00E+00	2 (0)
		$k_{G\_ed4}$	0.11	36.08 (48.17)	0.09	0.55	7.05E-03	59 (6)
134	12	$k_{G\_ed1}$	0.62	44.78 (49.91)	0.98	0.01	2.00E+00	3 (0)
		$k_{G\_ed2}$	0.62	44.78 (49.91)	1.00	0.00	7.05E-03	126 (0)
		$k_{G\_ed3}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	2 (0)
		$k_{G\_ed4}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	8 (1)
115	12	$k_{G\_ed1}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	3 (0)
		$k_{G\_ed2}$	0.63	46.09 (50.06)	1.00	0.00	7.05E-03	112 (0)
		$k_{G\_ed3}$	0.71	65.22 (47.84)	0.94	0.40	1.05E-04	86 (17)
		$k_{G\_ed4}$	0.63	59.13 (49.37)	0.77	0.44	2.00E+00	7 (1)
104	12	$k_{G\_ed1}$	0.67	63.46 (48.39)	0.78	0.51	1.05E-04	51 (3)
		$k_{G\_ed2}$	0.64	47.12 (50.16)	1.00	0.00	2.00E+00	100 (0)
		$k_{G\_ed3}$	0.64	47.12 (50.16)	1.00	0.00	2.00E+00	3 (0)
		$k_{G\_ed4}$	0.64	46.15 (50.09)	0.98	0.00	2.00E+00	10 (2)

Table A.12: MKL classification performance achieved by combining 12 edit kernels on Test Table

Size	Number	Kernel	F1	Acc	Sen	Spec	C	nSV
158	12	$k_{G\_ed1}$	0.97	97.47 (15.76)	0.94	1.00	3.25E-05	37 (4)
		$k_{G\_ed2}$	0.65	59.49 (49.25)	0.89	0.38	7.05E-03	111 (3)
		$k_{G\_ed3}$	0.25	25.95 (43.98)	0.30	0.23	2.00E+00	2 (0)

Therapy Table

Table A.13 –

Size	Number	Kernel	F1	Acc	Sen	Spec	C	nSV
		$k_{G\_ed4}$	0.94	94.94 (21.99)	1.00	0.91	7.05E-03	36 (5)
134	12	$k_{G\_ed1}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	2 (0)
		$k_{G\_ed2}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	28 (3)
		$k_{G\_ed3}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	2 (0)
		$k_{G\_ed4}$	0.62	44.78 (49.91)	1.00	0.00	2.00E+00	2 (0)
115	12	$k_{G\_ed1}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	2 (0)
		$k_{G\_ed2}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	21 (3)
		$k_{G\_ed3}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	2 (0)
		$k_{G\_ed4}$	0.63	46.09 (50.06)	1.00	0.00	2.00E+00	2 (0)
104	12	$k_{G\_ed1}$	0.64	47.12 (50.16)	1.00	0.00	2.00E+00	2 (0)
		$k_{G\_ed2}$	0.64	47.12 (50.16)	1.00	0.00	2.00E+00	18 (4)
		$k_{G\_ed3}$	0.64	47.12 (50.16)	1.00	0.00	2.00E+00	2 (0)
		$k_{G\_ed4}$	0.64	47.12 (50.16)	1.00	0.00	2.00E+00	2 (0)

Table A.13: MKL classification performance achieved by combining 12 edit kernels on Therapy Table

K	F1	Acc(%) $\pm$ (std)	Sen	Spec	Mode	Trans	C	nSV	%-ve Eig
<b>Pseudo</b>									
$k_{ed1}$	0.62	51.90 (50.12)	0.95	0.21	clip	Norm	128	22.84 (0.58)	69.62
$k_{ed2}$	0.59	50.63 (50.15)	0.85	0.26	flip	None	0.25	47.70 (0.66)	67.09
$k_{ed3}$	0.57	48.10 (50.12)	0.82	0.24	nsd	None	0.25	2.01 (0.16)	58.86
$k_{ed4}$	0.61	55.06 (49.90)	0.83	0.35	square	Norm	0.25	93.72 (1.39)	50.00
<b>Similarity</b>									
$k_{s\_ed1}$	0.48	52.53 (50.09)	0.52	0.53	square	Norm	128	108.66 (0.64)	5.06
$k_{s\_ed2}$	0.60	44.30 (49.83)	0.98	0.05	square	Norm	0.25	42.08 (0.91)	20.25
$k_{s\_ed3}$	0.52	58.86 (49.37)	0.53	0.63	square	Norm	0.25	70.80 (0.61)	12.03
$k_{s\_ed4}$	0.60	51.90 (50.12)	0.86	0.27	clip	Norm	0.25	25.77 (0.65)	50.63
<b>Gaussian</b>									
$k_{G\_ed1}$	0.60	63.29 (48.35)	0.65	0.62	square	Norm	0.25	28.83 (0.49)	29.11
$k_{G\_ed2}$	0.61	50.00 (50.16)	0.94	0.18	clip	None	128	2.00 (0.00)	32.28
$k_{G\_ed3}$	0.59	44.94 (49.90)	0.94	0.10	flip	Norm	128	9.94 (0.40)	40.51
$k_{G\_ed4}$	0.62	53.80 (50.01)	0.89	0.28	square	None	0.25	16.92 (0.32)	50.00
<b>Rat Quad</b>									
$k_{r\_ed1}$	0.56	54.43 (49.96)	0.68	0.45	square	None	0.25	56.89 (0.67)	17.09
$k_{r\_ed2}$	0.61	41.77 (49.48)	0.68	0.45	square	Norm	0.25	4.00 (0.00)	32.28
$k_{r\_ed3}$	0.59	42.41 (49.58)	1.00	0.01	flip	Norm	128	6.85 (0.80)	40.51
$k_{r\_ed4}$	<b>0.63</b>	<b>55.06 (49.90)</b>	<b>0.91</b>	<b>0.29</b>	<b>nsd</b>	<b>None</b>	<b>0.25</b>	<b>38.85 (0.44)</b>	<b>50.63</b>
<b>Poly</b>									
$k_{p\_ed1}$	0.57	46.84 (50.06)	0.86	0.18	flip	Norm	0.25	110.32 (0.56)	4.43
$k_{p\_ed2}$	0.59	45.57 (49.96)	0.94	0.11	flip	Norm	0.25	49.78 (1.01)	15.82
$k_{p\_ed3}$	0.59	43.67 (49.76)	0.98	0.04	square	Norm	0.25	46.02 (0.86)	9.49
$k_{p\_ed4}$	0.62	54.43 (49.96)	0.86	0.32	nsd	None	0.0005	107.49 (0.87)	50.00

Table A.14: Table showing results obtained by executing the kernel functions on a pair of sequences with common symbols



## Appendix B

# Sample Raw Data Extracts

### B.1 Clinical Table

No:	Patient ID	Event date	Read Code	Read Term
1	ID6568	19960412	66A..00	Diabetic monitoring
2	ID6568	19960412	6872	Diabetes mellitus screen
3	ID6568	19960412	6781	Health education offered
4	ID6568	19960412	6673	Driving licence
5	ID6504	19960412	H05z.00	Upper respiratory infect.NOS
6	ID6280	19960412	H06z011	Chest infection
7	ID5587	19960412	14L..00	H/O: drug allergy
8	ID5060	19960412	9OW4.00	New patient screen 1st letter
9	ID4906	19960412	A52..00	Chickenpox - varicella
10	ID4068	19960412	246..00	O/E - blood pressure reading
11	ID4068	19960412	4K22.00	Cervical smear: negative
12	ID2381	19960412	246..00	O/E - blood pressure reading
13	ID2371	19960412	ZV25015	[V]Oral contracept.prescriptn.
14	ID2371	19960412	246..00	O/E - blood pressure reading
15	ID2313	19960412	62...00	Patient pregnant
16	ID2257	19960412	13O5.00	Attendance allowance
17	ID2284	19960412	9OW4.00	New patient screen 1st letter
18	ID1954	19960412	9OG1.00	Geriatric screen - seen
19	ID1953	19960412	663..11	Asthma monitoring
20	ID1737	19960412	662..00	Cardiac disease monitoring
21	ID1737	19960412	6781	Health education offered
22	ID1737	19960412	6781	Health education offered
23	ID1390	19960412	H01..00	Acute sinusitis
24	ID853	19960412	J025000	Dental abscess
25	ID897	19960412	A52..00	Chickenpox - varicella
26	ID411	19960412	ZV25015	[V]Oral contracept.prescriptn.
27	ID411	19960412	246..00	O/E - blood pressure reading
28	ID340	19960412	246..00	O/E - blood pressure reading
29	ID236	19960412	H06z011	Chest infection
30	ID142	19960412	246..00	O/E - blood pressure reading
31	ID2174	19960412	H01..11	Sinusitis
32	ID9391	19960413	J43..11	Gastroenteritis
33	ID4535	19960413	J43..11	Gastroenteritis
34	ID4535	19960413	246..00	O/E - blood pressure reading

Clinical Table

Table B.1 –Table showing a few rows from the Clinical table

No:	Patient ID	Event date	Read Code	Read Term
35	ID3979	19960413	246..00	O/E - blood pressure reading
36	ID3086	19960413	L166z11	UTI in pregnancy
37	ID571	19960413	H141.12	Enlargement - tonsil/adenoid
38	ID9449	19960414	F48y000	Blurred vision NOS
39	ID9041	19960414	R002300	[D]Collapse
40	ID8862	19960414	5353	Standard chest X-ray abnormal
41	ID8000	19960414	BBE0.12	[M]Naevus NOS
42	ID9451	19960415	9OW1.00	Attended new patient screen
43	ID9451	19960415	1374	Moderate smoker - 10-19 cigs/d
44	ID9451	19960415	229..00	O/E - height
45	ID9451	19960415	22A..00	O/E - weight
46	ID9451	19960415	246..00	O/E - blood pressure reading
47	ID9451	19960415	115..00	No significant medical history
48	ID9451	19960415	1226	No FH: Ischaemic heart disease
49	ID9451	19960415	1225.11	No FH: CVA/Stroke/TIA
50	ID9451	19960415	138..00	Exercise grading
51	ID9451	19960415	663..11	Asthma monitoring
52	ID9451	19960415	13A..00	Diet - patient initiated
53	ID9451	19960415	1221	No FH: Glaucoma
54	ID9451	19960415	679..11	Advice to patient - subject
55	ID9451	19960415	6781	Health education offered
56	ID9451	19960415	679..11	Advice to patient - subject
57	ID9451	19960415	6781	Health education offered
58	ID9449	19960415	H51z.00	Pleural effusion NOS
59	ID9041	19960415	G66..00	Stroke/CVA unspecified
60	ID9041	19960415	G6...00	Cerebrovascular disease
61	ID9041	19960415	94...00	Death administration
62	ID8679	19960415	662..00	Cardiac disease monitoring
63	ID8679	19960415	6781	Health education offered
64	ID8679	19960415	6781	Health education offered
65	ID8505	19960415	M113.00	Flexural eczema
66	ID8378	19960415	H01..00	Acute sinusitis
67	ID8378	19960415	115..00	No significant medical history
68	ID8356	19960415	H060.00	Acute bronchitis
69	ID8372	19960415	246..00	O/E - blood pressure reading
70	ID7910	19960415	13M..00	Family bereavement
71	ID7594	19960415	H01..00	Acute sinusitis
72	ID7251	19960415	G800500	Thromboph.superf.leg vein NOS
73	ID7013	19960415	4K23.00	Cerv.smear: mild dyskaryosis
74	ID6890	19960415	F504.11	Wax in ear
75	ID6693	19960415	614D.00	Oral contraceptive prescribed

Table B.1: Clinical Table

## B.2 Recall Table

No:	Patient ID	Event date	Read Code	Read Term
1	ID315	19940620	5372	Mammography normal
2	ID314	19941214	4K22.00	Cervical smear: negative
3	ID315	19950203	246..00	O/E - blood pressure reading
4	ID315	19950425	246..00	O/E - blood pressure reading
5	ID314	19950928	617..11	Sheath contraception
6	ID315	19951120	246..00	O/E - blood pressure reading
7	ID314	19960610	4K22.00	Cervical smear: negative
8	ID314	19960828	61...00	Contraception
9	ID314	19960828	614D.00	Oral contraceptive prescribed
10	ID315	19970113	246..00	O/E - blood pressure reading
11	ID315	19970421	4K22.00	Cervical smear: negative
12	ID314	19970818	4K22.00	Cervical smear: negative
13	ID314	19971016	617..11	Sheath contraception
14	ID314	19981203	6148	Progestagen only oral contrac.
15	ID315	19981214	246..00	O/E - blood pressure reading
16	ID314	20000505	246..00	O/E - blood pressure reading
17	ID314	20000525	ZV25000	[V]General contracept. advice
18	ID314	20000728	4K28.00	Cerv.smear: mod.dyskaryosis
19	ID314	20001228	4K22.00	Cervical smear: negative
20	ID314	20010611	4K22.00	Cervical smear: negative
21	ID314	20010611	4K22.00	Cervical smear: negative
22	ID314	20011218	4K22.00	Cervical smear: negative
23	ID317	19800506	4K29.00	Cerv.smear: borderline changes
24	ID322	19861006	6151	IUD fitted
25	ID322	19871116	6154	IUD checked - no problems
26	ID322	19871216	4K22.00	Cervical smear: negative
27	ID322	19880824	246..00	O/E - blood pressure reading
28	ID317	19880923	4K22.00	Cervical smear: negative
29	ID322	19881102	6154	IUD checked - no problems
30	ID322	19900302	6154	IUD checked - no problems
31	ID322	19900425	4K22.00	Cervical smear: negative
32	ID322	19910920	6153	IUD re-fitted
33	ID322	19920610	6154	IUD checked - no problems
34	ID322	19920707	246..00	O/E - blood pressure reading
35	ID317	19920923	4K22.00	Cervical smear: negative
36	ID317	19921027	246..00	O/E - blood pressure reading
37	ID322	19930428	6154	IUD checked - no problems
38	ID322	19930428	4K22.00	Cervical smear: negative
39	ID317	19931110	246..00	O/E - blood pressure reading
40	ID322	19940607	6862	Breast neoplasm screen
41	ID317	19940607	537..11	Mammography - X-ray
42	ID316	19940613	246..00	O/E - blood pressure reading
43	ID316	19940811	537..11	Mammography - X-ray
44	ID322	19960501	246..00	O/E - blood pressure reading
45	ID322	19960501	4K22.00	Cervical smear: negative
46	ID322	19961023	246..00	O/E - blood pressure reading
47	ID322	19970730	246..00	O/E - blood pressure reading

Recall Table

Table B.2 –Table showing a few rows from the Recall table

No:	Patient ID	Event date	Read Code	Read Term
48	ID316	19980316	246..00	O/E - blood pressure reading
49	ID317	19980612	246..00	O/E - blood pressure reading
50	ID320	19980615	4K22.00	Cervical smear: negative
51	ID316	19981105	246..00	O/E - blood pressure reading
52	ID322	19981125	7E09.00	IUCD procedure
53	ID316	19990506	246..00	O/E - blood pressure reading
54	ID317	19990727	246..00	O/E - blood pressure reading
55	ID322	19991103	6859	Ca cervix - screen done
56	ID322	19991103	6859	Ca cervix - screen done
57	ID320	20000106	246..00	O/E - blood pressure reading
58	ID316	20000121	246..00	O/E - blood pressure reading
59	ID320	20000619	246..00	O/E - blood pressure reading
60	ID320	20000807	61A..11	Morning-after pill
61	ID316	20010906	246..00	O/E - blood pressure reading
62	ID335	19740201	6151	IUD fitted
63	ID333	19850530	4K22.00	Cervical smear: negative
64	ID335	19851209	4K22.00	Cervical smear: negative
65	ID325	19860407	246..00	O/E - blood pressure reading
66	ID329	19861119	4K22.00	Cervical smear: negative
67	ID323	19880129	246..00	O/E - blood pressure reading
68	ID333	19890109	246..00	O/E - blood pressure reading
69	ID335	19890512	4K22.00	Cervical smear: negative
70	ID335	19890518	246..00	O/E - blood pressure reading
71	ID335	19890518	68...11	Screening - health check
72	ID333	19890731	246..00	O/E - blood pressure reading
73	ID323	19891016	6853	Ca cervix screen - not wanted
74	ID329	19891021	614D.00	Oral contraceptive prescribed
75	ID329	19891021	246..00	O/E - blood pressure reading

Table B.2: Recall Table

### B.3 Refer Table

No:	Patient ID	Event date	Read Code	Read Term
1	ID19	19850000	F59z.00	DEAF
2	ID23	19900411	4J...00	Microbiology
3	ID23	19901005	N14z.00	BACK PAIN
4	ID11	19910218	N091.00	HAEMARTHROSIS
5	ID11	19920102	F340.00	CARPAL TUNNEL SYNDROME
6	ID23	19920115	N06z.00	ARTHROPATHY
7	ID23	19920427	5372	Mammography normal
8	ID9	19921019	N142.11	LOW BACK PAIN
9	ID23	19930305	F340.00	CARPAL TUNNEL SYNDROME
10	ID23	19930625	19F..11	DIARRHOEA
11	ID23	19931115	19F..11	DIARRHOEA

Refer Table

Table B.3 –Table showing a few rows from the Refer table

No:	Patient ID	Event date	Read Code	Read Term
12	ID23	19940628	5372	Mammography normal
13	ID23	19950224	K5A2011	MENOPAUSAL HOT FLUSHES
14	ID23	19950711	N094611	PAIN KNEE
15	ID23	19951107	F4Kz411	RED EYE
16	ID13	19960409	773A100	Drainage of perianal abscess
17	ID13	19970403	N143.00	Sciatica
18	ID13	19971015	M12..11	Contact dermatitis
19	ID23	19980303	M2z0.00	Skin lesion
20	ID23	19990824	195..00	Indigestion symptoms
21	ID23	19990907	N217900	Plantar fasciitis
22	ID13	19991203	1AZ2.11	Infertility problem
23	ID23	19991217	7E01400	Avulsion of cervical polyp
24	ID23	20000107	K190.11	Recurrent urinary tract infect
25	ID23	20000211	6862.11	Mammography - screening
26	ID13	20000404	N131.00	Cervicalgia - pain in neck
27	ID13	20000829	S570400	Whiplash injury
28	ID23	20001205	7N15000	[SO]Conjunctiva
29	ID23	20010206	195..00	Indigestion symptoms
30	ID23	20010330	7N60.00	[SO]Vagina
31	ID16	20010717	K551.00	Dysplasia of cervix uteri
32	ID28	19940418	S....00	TRAUMA
33	ID28	19940627	537..11	Mammography - X-ray
34	ID28	19961223	26B7.00	O/E - shotty breast
35	ID28	19990413	N350.00	Hallux valgus - acquired
36	ID28	19990901	1594	H/O: genital prolapse
37	ID28	20000229	6862.11	Mammography - screening
38	ID28	20001101	1B5..11	Dizziness symptom
39	ID28	20010912	N245.13	Foot pain
40	ID28	20020103	S339.00	Fracture of fibula alone
41	ID25	19900522	F59z.00	DEAF
42	ID36	19901011	4J...00	Microbiology
43	ID26	19901031	F4F3000	DACRYOCYSTITIS
44	ID25	19911025	4J...00	Microbiology
45	ID34	19920117	4J...00	Microbiology
46	ID35	19920305	4J...00	Microbiology
47	ID34	19920903	1A59.00	PELVIC PAIN
48	ID36	19921023	4J...00	Microbiology
49	ID36	19921023	3395.13	Peak flow rate
50	ID25	19930315	SKz..00	INJURY
51	ID35	19930819	8H7B.00	REFERRED TO COMMUNITY PSYCHIATRIC NURSE
52	ID25	19931123	4J...00	Microbiology
53	ID26	19940620	E2B..00	DEPRESSION
54	ID26	19940620	5372	Mammography normal
55	ID27	19940819	8H7B.00	REFERRED TO COMMUNITY PSYCHIATRIC NURSE
56	ID27	19940822	K197.00	HAEMATURIA
57	ID27	19941123	R082.00	RETENTION URINE
58	ID30	19941129	N143.00	SCIATICA
59	ID35	19950110	8H7B.00	REFERRED TO COMMUNITY PSYCHIATRIC NURSE

Refer Table



Table B.3 –Table showing a few rows from the Refer table

No:	Patient ID	Event date	Read Code	Read Term
60	ID35	19950210	E00z.00	DEMENTIA
61	ID27	19950223	R002300	COLLAPSE
62	ID30	19950301	N145.00	PAIN BACK
63	ID25	19950704	G581.00	LVF (LEFT VENTRICULAR FAILURE)
64	ID25	19960322	194..11	Dysphagia
65	ID30	19960419	1D13.11	Pain
66	ID30	19960520	N143.00	Sciatica
67	ID25	19960713	173..13	Shortness of breath symptom
68	ID25	19970318	16ZZ.00	General symptom NOS
69	ID30	19981223	N143.00	Sciatica
70	ID25	19990102	H06z011	Chest infection
71	ID26	19990219	195..00	Indigestion symptoms
72	ID36	19990525	4J...00	Microbiology
73	ID26	19990603	4J...00	Microbiology
74	ID26	20000211	6862.11	Mammography - screening
75	ID26	20000229	6862.11	Mammography - screening

Table B.3: Refer Table

## B.4 Repeat Table

No:	Patient ID	Event date	Code	Name	Form	Strength	Qty
1	ID8487	19880419	53647020	ZOVIRAX CRE 5	CRE	5	2
2	ID8487	19880419	52575020	ADALAT RETARD TAB 20	TAB	20	60
3	ID8487	19880419	53647020	ZOVIRAX crm 5%	CRE	5	2
4	ID9289	19880419	69640020	PREDFOAM AEROSOL REC 20		20	2
5	ID9289	19880419	83297020	SALAZOPYRIN TAB 500	TAB	500	100
6	ID9289	19880419	69640020	PREDFOAM rectal foam 20mg / dose	FOA	20	2
7	ID9369	19880419	54815020	TENORETIC TAB	TAB	0	56
8	ID668	19880420	58502020	HALDOL LIQ 2	LIQ	2	100
9	ID668	19880420	50020020	KEMADRIN TAB 5	TAB	5	126
10	ID668	19880420	58502020	HALDOL sf liq 2mg/ml	LIQ	2	100
11	ID668	19880420	50020020	KEMADRIN tabs 5mg	TAB	5	126
12	ID1388	19880420	66877020	TEMAZEPAM CAP 10	CAP	10	30
13	ID1388	19880420	66877020	TEMAZEPAM caps 10mg	CAP	10	30
14	ID1635	19880420	49976020	ISOGEL GRA	GRA	0	400
15	ID1635	19880420	62690020	PROPINE EYE DRO 0.1	DRO	0.1	10
16	ID1635	19880420	51942020	TIMOPTOL EYE DRO 0.5	DRO	0.5	10
17	ID1654	19880420	54461020	BETNOVATE CRE 0.1	CRE	0.1	30
18	ID1654	19880420	60153020	ATENOLOL TAB 50	TAB	50	28
19	ID1654	19880420	54461020	BETNOVATE crm 0.1%	CRE	0.1	30
20	ID1654	19880420	60153020	ATENOLOL tabs 50mg	TAB	50	28
21	ID3873	19880420	51584020	SLOW-TRASICOR TAB 160	TAB	160	28
22	ID4582	19880420	66877020	TEMAZEPAM CAP 10	CAP	10	28
23	ID4582	19880420	68517020	ZANTAC TAB 150	TAB	150	30

Repeat Table

Table B.4 –Table showing a few rows from the Repeat table

No:	Patient ID	Event date	Code	Name	Form	Strength	Qty
24	ID4582	19880420	60972020	CO-CODAMOL EFFERVESC TAB	TAB	0	100
25	ID4582	19880420	51541020	SERC TAB 8	TAB	8	120
26	ID5472	19880420	51745020	SURGAM TAB 200	TAB	200	28
27	ID5472	19880420	55991020	THYROXINE TAB 100	TAB	100	28
28	ID5472	19880420	55883020	HYPROMELLOSE EYE DRO 0.3	DRO	0.3	10
29	ID5472	19880420	54196020	PLAQUENIL TAB 200	TAB	200	14
30	ID5472	19880420	55991020	THYROXINE tabs 100micrograms	TAB	100	28
31	ID5959	19880420	60153020	ATENOLOL TAB 50	TAB	50	28
32	ID5959	19880420	60153020	ATENOLOL tabs 50mg	TAB	50	28
33	ID6212	19880420	52399020	BENORAL TAB 750	TAB	750	1
34	ID6212	19880420	56363020	GASTROCOTE TAB	TAB	0	100
35	ID7239	19880420	59841020	OXAZEPAM TAB 15	TAB	15	100
36	ID7239	19880420	50841020	OILATUM EMOLLIENT LIQ	LIQ	0	350
37	ID7239	19880420	58922020	NAPROXEN TAB 250	TAB	250	60
38	ID7239	19880420	60972020	CO-CODAMOL EFFERVESC TAB	TAB	0	100
39	ID7239	19880420	55991020	THYROXINE TAB 100	TAB	100	30
40	ID7239	19880420	59841020	OXAZEPAM tabs 15mg	TAB	15	100
41	ID7239	19880420	58922020	NAPROXEN tabs 250mg	TAB	250	60
42	ID7239	19880420	55991020	THYROXINE tabs 100micrograms	TAB	100	30
43	ID7241	19880420	52717020	STEMETIL SUP 25	SUP	25	10
44	ID7241	19880420	60972020	CO-CODAMOL EFFERVESC TAB	TAB	0	100
45	ID7241	19880420	55991020	THYROXINE TAB 100	TAB	100	30
46	ID7241	19880420	59355020	IBUPROFEN TAB 400	TAB	400	90
47	ID7241	19880420	56015020	QUININE SULPHATE TAB 300	TAB	300	30
48	ID7241	19880420	58928020	NITRAZEPAM TAB 5	TAB	5	30
49	ID7241	19880420	51690020	STEMETIL TAB 5	TAB	5	84
50	ID7241	19880420	66455020	RANITIDINE TAB 150	TAB	150	30
51	ID7241	19880420	55991020	THYROXINE tabs 100micrograms	TAB	100	30
52	ID7241	19880420	59355020	IBUPROFEN tabs 400mg	TAB	400	90
53	ID7241	19880420	56015020	QUININE SULPHATE tabs 300mg	TAB	300	30
54	ID7241	19880420	58928020	NITRAZEPAM tabs 5mg	TAB	5	30
55	ID7241	19880420	66455020	RANITIDINE tabs 150mg	TAB	150	30
56	ID7529	19880420	67266020	BECOTIDE 200 INH 200	INH	200	1
57	ID7641	19880420	48833020	DAKTARIN CRE 2	CRE	2	15
58	ID7726	19880420	68517020	ZANTAC TAB 150	TAB	150	60
59	ID7906	19880420	60153020	ATENOLOL TAB 50	TAB	50	28
60	ID7906	19880420	60153020	ATENOLOL tabs 50mg	TAB	50	28
61	ID8584	19880420	49223020	ELTROXIN TAB 100	TAB	100	30
62	ID8584	19880420	55403020	TOFRANIL TAB 25	TAB	25	100
63	ID8584	19880420	55990020	THYROXINE TAB 50	TAB	50	30
64	ID8584	19880420	58922020	NAPROXEN TAB 250	TAB	250	100
65	ID8734	19880420	56015020	QUININE SULPHATE TAB 300	TAB	300	30
66	ID8734	19880420	53107020	FRUMIL TAB	TAB	0	56
67	ID8734	19880420	49925020	IMODIUM CAP 2	CAP	2	30
68	ID8734	19880420	54963020	BONJELA JEL	JEL	0	15
69	ID8734	19880420	51474020	SANDOCAL 1000 TAB 1000	TAB	1000	30
70	ID8734	19880420	58747020	TOPAL TAB	TAB	0	42
71	ID8734	19880420	58932020	PARACETAMOL TAB 500	TAB	500	60

Repeat Table

Table B.4 –Table showing a few rows from the Repeat table

No:	Patient ID	Event date	Code	Name	Form	Strength	Qty
72	ID8734	19880420	56015020	QUININE SULPHATE tabs 300mg	TAB	300	30
73	ID8734	19880420	49925020	IMODIUM caps 2mg	CAP	2	30
74	ID8734	19880420	54963020	BONJELA gel	GEL	0	15
75	ID8734	19880420	65685020	Glutafin GF WF biscuit(s) [NUTRICIA]	BIS	0	4

Table B.4: Repeat Table

## B.5 Test Table

No:	Patient ID	Event date	Read Code	Read Term	Value	Normal Range	
						Min	Max
1	ID8329	20000525	423..00	Haemoglobin estimation	14.4	11.5	16
2	ID8329	20000525	425..00	Haematocrit - PCV	0.43	0.36	0.46
3	ID8329	20000525	426..00	Red blood cell (RBC) count	4.89	4	5.2
4	ID8329	20000525	428..00	Mean corpusc. haemoglobin(MCH)	29.4	25	35
5	ID8329	20000525	429..00	Mean corpusc. Hb. conc. (MCHC)	33.3	31	36
6	ID8329	20000525	42A..00	Mean corpuscular volume (MCV)	88.3	80	100
7	ID8329	20000525	42B6.00	Erythrocyte sedimentation rate	19	1	12
8	ID8329	20000525	42H..00	Total white cell count	8.46	4	10.5
9	ID8329	20000525	42J..00	Neutrophil count	5.15	1.8	7.5
10	ID8329	20000525	42K..00	Eosinophil count	0.12	0.02	0.4
11	ID8329	20000525	42L..00	Basophil count	0.04	0	0.2
12	ID8329	20000525	42M..00	Lymphocyte count	2.66	1.5	4
13	ID8329	20000525	42N..00	Monocyte count	0.49	0.2	0.8
14	ID8329	20000525	42P..00	Platelet count	270	145	400
15	ID8329	20000525	4427	Free T4 level	16.3	10	26
16	ID8329	20000525	442A.00	TSH - thyroid stim. hormone	1.6	0.25	5.5
17	ID8329	20000525	44E..00	Serum bilirubin level	11	3	22
18	ID8329	20000525	44F..00	Serum alkaline phosphatase	97	38	126
19	ID8329	20000525	44G3.00	ALT/SGPT serum level	43	7	56
20	ID8329	20000525	44I4.00	Serum potassium	4.9	3.6	5.3
21	ID8329	20000525	44I5.00	Serum sodium	140	134	145
22	ID8329	20000525	44IC.00	Corrected serum calcium level	2.31	2.1	2.55
23	ID8329	20000525	44J3.00	Serum creatinine	87	62	133
24	ID8329	20000525	44M4.00	Serum albumin	46	35	49
25	ID419	20000526	423..00	Haemoglobin estimation	15.7	11.5	16
26	ID419	20000526	425..00	Haematocrit - PCV	0.5	0.36	0.46
27	ID419	20000526	426..00	Red blood cell (RBC) count	5.2	4	5.2
28	ID419	20000526	428..00	Mean corpusc. haemoglobin(MCH)	30.2	25	35
29	ID419	20000526	429..00	Mean corpusc. Hb. conc. (MCHC)	31.7	31	36
30	ID419	20000526	42A..00	Mean corpuscular volume (MCV)	95.4	80	100
31	ID419	20000526	42H..00	Total white cell count	9.04	4	10.5
32	ID419	20000526	42J..00	Neutrophil count	4.94	1.8	7.5
33	ID419	20000526	42K..00	Eosinophil count	0.23	0.02	0.4
34	ID419	20000526	42L..00	Basophil count	0.04	0	0.2

Test Table

Table B.5 –Table showing a few rows from the Test table

No:	Patient ID	Event date	Read Code	Read Term	Value	Normal Range	
						Min	Max
35	ID419	20000526	42M..00	Lymphocyte count	3.41	1.5	4
36	ID419	20000526	42N..00	Monocyte count	0.42	0.2	0.8
37	ID419	20000526	42P..00	Platelet count	165	145	400
38	ID419	20000526	44E..00	Serum bilirubin level	9	3	22
39	ID419	20000526	44F..00	Serum alkaline phosphatase	72	38	126
40	ID419	20000526	44G3.00	ALT/SGPT serum level	22	7	56
41	ID419	20000526	44I4.00	Serum potassium	4.1	3.6	5.3
42	ID419	20000526	44I5.00	Serum sodium	142	134	145
43	ID419	20000526	44IC.00	Corrected serum calcium level	2.39	2.1	2.55
44	ID419	20000526	44J3.00	Serum creatinine	81	62	133
45	ID419	20000526	44M4.00	Serum albumin	38	35	49
46	ID419	20000526	44P..00	Serum cholesterol	3.9	3	6.5
47	ID441	20000526	44I4.00	Serum potassium	4.4	3.6	5.3
48	ID441	20000526	44I5.00	Serum sodium	134	134	145
49	ID441	20000526	44J3.00	Serum creatinine	80	62	133
50	ID441	20000526	44P..00	Serum cholesterol	6	3	6.5
51	ID3544	20000526	423..00	Haemoglobin estimation	13.5	11.5	16
52	ID3544	20000526	425..00	Haematocrit - PCV	0.41	0.36	0.46
53	ID3544	20000526	426..00	Red blood cell (RBC) count	4.68	4	5.2
54	ID3544	20000526	428..00	Mean corpusc. haemoglobin(MCH)	28.8	25	35
55	ID3544	20000526	429..00	Mean corpusc. Hb. conc. (MCHC)	33.3	31	36
56	ID3544	20000526	42A..00	Mean corpuscular volume (MCV)	86.5	80	100
57	ID3544	20000526	42B6.00	Erythrocyte sedimentation rate	1	1	12
58	ID3544	20000526	42H..00	Total white cell count	7.04	4	10.5
59	ID3544	20000526	42J..00	Neutrophil count	3.72	1.8	7.5
60	ID3544	20000526	42K..00	Eosinophil count	0.19	0.02	0.4
61	ID3544	20000526	42L..00	Basophil count	0.11	0	0.2
62	ID3544	20000526	42M..00	Lymphocyte count	2.68	1.5	4
63	ID3544	20000526	42N..00	Monocyte count	0.34	0.2	0.8
64	ID3544	20000526	42P..00	Platelet count	235	145	400
65	ID3544	20000526	4427	Free T4 level	13.5	10	26
66	ID3544	20000526	442A.00	TSH - thyroid stim. hormone	1.1	0.25	5.5
67	ID3544	20000526	44E..00	Serum bilirubin level	9	3	22
68	ID3544	20000526	44F..00	Serum alkaline phosphatase	57	38	126
69	ID3544	20000526	44G3.00	ALT/SGPT serum level	26	7	56
70	ID3544	20000526	44I4.00	Serum potassium	4.4	3.6	5.3
71	ID3544	20000526	44I5.00	Serum sodium	139	134	145
72	ID3544	20000526	44IC.00	Corrected serum calcium level	2.4	2.1	2.55
73	ID3544	20000526	44J3.00	Serum creatinine	67	62	133
74	ID3544	20000526	44M4.00	Serum albumin	44	35	49
75	ID4186	20000526	423..00	Haemoglobin estimation	15.4	12.5	18

Table B.5: Test Table

## B.6 Therapy Table

No:	Patient ID	Event date	Code	Drug Name	Form	Strength	Qty
1	ID5535	19880427	5271007	BISACODYL TAB 10	TAB	10	56
2	ID5562	19880427	49223020	ELTROXIN TAB 100	TAB	100	100
3	ID5562	19880427	51609020	GAVISCON TAB	TAB	0	180
4	ID5562	19880427	53819020	DELTACORTRIL ENTERIC TAB 2.5	TAB	2.5	100
5	ID5562	19880427	53647020	ZOVIRAX CRE 5	CRE	5	1
6	ID5562	19880427	49223020	ELTROXIN TAB 100	TAB	100	100
7	ID5562	19880427	53819020	DELTACORTRIL ENTERIC TAB 2.5	TAB	2.5	100
8	ID5562	19880427	49223020	ELTROXIN TAB 100	TAB	100	100
9	ID5562	19880427	53819020	DELTACORTRIL ENTERIC TAB 2.5	TAB	2.5	100
10	ID5702	19880427	58922020	NAPROXEN TAB 250	TAB	250	100
11	ID6001	19880427	57784020	MINOCIN 50 TAB 50	TAB	50	84
12	ID6001	19880427	70870020	HYDROCORTISONE CRE 0.1	CRE	0.1	15
13	ID6001	19880427	57784020	MINOCIN 50 TAB 50	TAB	50	84
14	ID6380	19880427	52609020	ACEPRIL TAB 25	TAB	25	56
15	ID7028	19880427	66580020	SENNA TAB	TAB	0	56
16	ID7399	19880427	59420020	FRUSEMIDE TAB 40	TAB	40	28
17	ID7399	19880427	53557020	LANOXIN-125 TAB 125	TAB	125	28
18	ID8733	19880427	53107020	FRUMIL TAB	TAB	0	56
19	ID8733	19880427	52213020	VENTOLIN INH	INH	0	1
20	ID132	19880428	59420020	FRUSEMIDE TAB 40	TAB	40	56
21	ID132	19880428	49981020	ISORDIL TAB 10	TAB	10	100
22	ID132	19880428	68517020	ZANTAC TAB 150	TAB	150	60
23	ID132	19880428	58897020	CO-PROXAMOL TAB	TAB	0	100
24	ID132	19880428	50091020	LANOXIN TAB 250	TAB	250	28
25	ID469	19880428	59354020	IBUPROFEN TAB 200	TAB	200	56
26	ID821	19880428	53298020	VOLTAROL RETARD TAB 100	TAB	100	28
27	ID1080	19880428	54818020	TENORMIN L.S. CALEND TAB 50	TAB	50	28
28	ID1080	19880428	59354020	IBUPROFEN TAB 200	TAB	200	84
29	ID1080	19880428	51239020	PREMARIN TAB 625	TAB	625	21
30	ID1374	19880428	54818020	TENORMIN L.S. TAB 50	TAB	50	28
31	ID1940	19880428	66877020	TEMAZEPAM CAP 10	CAP	10	14
32	ID2743	19880428	51482020	SANOMIGRAN TAB 1.5	TAB	1.5	28
33	ID2877	19880428	57757020	NAPROSYN TAB 250	TAB	250	56
34	ID2877	19880428	58943020	AMILORIDE TAB 5	TAB	5	56
35	ID3831	19880428	54493020	INDERAL TAB 10	TAB	10	180
36	ID4070	19880428	54886020	TRASICOR TAB 80	TAB	80	60
37	ID4070	19880428	59420020	FRUSEMIDE TAB 40	TAB	40	60
38	ID4070	19880428	51581020	SLOW-K TAB 600	TAB	600	60
39	ID4070	19880428	54886020	TRASICOR TAB 80	TAB	80	60
40	ID4070	19880428	51816020	TAGAMET TAB 400	TAB	400	60
41	ID4070	19880428	53290020	VOLTAROL SUP 100	SUP	100	30
42	ID4070	19880428	48333020	BENEMID TAB 500	TAB	500	60
43	ID4434	19880428	56363020	GASTROCOTE TAB	TAB	0	200
44	ID4434	19880428	52400020	BENORAL SUS 40	SUS	40	300
45	ID4434	19880428	56363020	GASTROCOTE TAB	TAB	0	200
46	ID4564	19880428	54359020	ASPIRIN TAB 300	TAB	300	60
47	ID4709	19880428	58977020	BENDROFLUAZIDE TAB 5	TAB	5	20

Therapy Table

Table B.6 –Table showing a few rows from the Therapy table

No:	Patient ID	Event date	Code	Name	Form	Strength	Qty
48	ID4709	19880428	50280020	MAGNAPEN CAP 500	CAP	500	28
49	ID4875	19880428	54570020	TEGRETOL TAB 200	TAB	200	60
50	ID4875	19880428	58982020	DIAZEPAM TAB 10	TAB	10	30
51	ID5564	19880428	2815007	COCONUT OIL COMPOUND OIN	OIN	0	450
52	ID5675	19880428	56363020	GASTROCOTE TAB	TAB	0	200
53	ID5675	19880428	51860020	TENORMIN TAB 100	TAB	100	28
54	ID6248	19880428	59354020	IBUPROFEN TAB 200	TAB	200	100
55	ID6248	19880428	68342020	TEMAZEPAM TAB 10	TAB	10	30
56	ID6321	19880428	52213020	VENTOLIN INH	INH	0	2
57	ID6321	19880428	54461020	BETNOVATE CRE 0.1	CRE	0.1	15
58	ID6321	19880428	57212020	BECOTIDE 100 INH 100	INH	100	1
59	ID6321	19880428	52213020	VENTOLIN INH	INH	0	2
60	ID6586	19880428	59354020	IBUPROFEN TAB 200	TAB	200	90
61	ID7558	19880428	54471020	BETNOVATE RD OIN 0.02	OIN	0.02	100
62	ID7558	19880428	52011020	TRILUDAN TAB 60	TAB	60	30
63	ID7558	19880428	49882020	HYPOVASE TAB 2	TAB	2	56
64	ID8627	19880428	52575020	ADALAT RETARD TAB 20	TAB	20	100
65	ID8627	19880428	54815020	TENORETIC TAB	TAB	0	28
66	ID8666	19880428	54818020	TENORMIN L.S. TAB 50	TAB	50	28
67	ID9210	19880428	66580020	SENNA TAB	TAB	0	56
68	ID9210	19880428	54359020	ASPIRIN TAB 300	TAB	300	28
69	ID9210	19880428	62791020	DIOCTYL TAB 100	TAB	100	120
70	ID9210	19880428	54359020	ASPIRIN TAB 300	TAB	300	28
71	ID775	19880429	54818020	TENORMIN L.S. CALEND TAB 50	TAB	50	28
72	ID800	19880429	54818020	TENORMIN L.S. CALEND TAB 50	TAB	50	28
73	ID800	19880429	52574020	ADALAT RETARD 10 TAB 10	TAB	10	56
74	ID848	19880429	58922020	NAPROXEN TAB 250	TAB	250	50
75	ID848	19880429	58922020	NAPROXEN TAB 250	TAB	250	50

Table B.6: Therapy Table

# Appendix C

## Kernel Evaluation

Method	Distance	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
Pseudo		0	4	75.16 (25.65)	0.00 (0.00)	0.74 (0.32)
	$d_{ed1}$	0	1	99.37 (0.00)	0.00 (0.00)	0.93 (0.00)
	$d_{ed2}$	0	1	94.94 (0.00)	0.00 (0.00)	0.99 (0.00)
	$d_{ed3}$	0	1	56.96 (0.00)	0.00 (0.00)	0.75 (0.00)
	$d_{ed4}$	0	1	49.37 (0.00)	0.00 (0.00)	0.29 (0.00)
Similarity		2	2	25.63 (33.57)	6.35 (5.53)	0.67 (0.38)
	$d_{ed1}$	1	0	0.00 (0.00)	12.53 (0.00)	0.15 (0.00)
	$d_{ed2}$	0	1	1.90 (0.00)	2.16 (0.00)	0.99 (0.00)
	$d_{ed3}$	1	0	0.00 (0.00)	9.48 (0.00)	0.64 (0.00)
	$d_{ed4}$	0	1	49.37 (0.00)	1.24 (0.00)	0.90 (0.00)
Gaussian		8	12	31.59 (20.97)	4.65 (5.22)	0.70 (0.41)
	$d_{ed1}$	5	0	0.00 (0.00)	7.89 (5.90)	0.47 (0.48)
	$d_{ed2}$	1	4	3.04 (1.87)	3.75 (5.01)	0.81 (0.41)
	$d_{ed3}$	2	3	24.05 (22.14)	5.80 (6.19)	0.62 (0.49)
	$d_{ed4}$	0	0	48.73 (2.49)	1.16 (0.21)	0.92 (0.12)
Rat Quad		7	5	29.49 (25.02)	7.38 (5.23)	0.56 (0.37)
	$d_{ed1}$	3	0	0.00 (0.00)	12.53 (0.03)	0.13 (0.05)
	$d_{ed2}$	1	2	1.48 (1.32)	5.42 (5.84)	0.77 (0.38)
	$d_{ed3}$	3	0	0.00 (0.00)	10.22 (2.08)	0.48 (0.33)
	$d_{ed4}$	0	3	47.68 (3.49)	1.35 (0.20)	0.85 (0.10)
Poly		6	6	25.32 (25.89)	4.60 (3.63)	0.87 (0.18)
	$d_{ed1}$	3	0	0.00 (0.00)	7.37 (3.98)	0.77 (0.28)
	$d_{ed2}$	0	3	1.69 (0.37)	2.97 (1.38)	0.98 (0.02)
	$d_{ed3}$	3	0	0.00 (0.00)	6.79 (3.90)	0.81 (0.24)
	$d_{ed4}$	0	3	48.95 (0.73)	1.27 (0.05)	0.90 (0.04)

Table C.1: Static kernel assessment Clinical dataset

Method	Distance	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
Pseudo		0	4	68.99 (12.64)	0.00 (0.00)	0.83 (0.08)
	$d_{ed1}$	0	1	77.22 (0.00)	0.00 (0.00)	0.82 (0.00)
	$d_{ed2}$	0	1	82.28 (0.00)	0.00 (0.00)	0.94 (0.00)
	$d_{ed3}$	0	1	59.49 (0.00)	0.00 (0.00)	0.79 (0.00)
	$d_{ed4}$	0	1	56.96 (0.00)	0.00 (0.00)	0.77 (0.00)
Similarity		0	4	7.91 (1.51)	4.93 (2.41)	0.73 (0.19)
	$d_{ed1}$	0	1	6.96 (0.00)	6.89 (0.00)	0.57 (0.00)
	$d_{ed2}$	0	1	10.13 (0.00)	1.62 (0.00)	0.99 (0.00)
	$d_{ed3}$	0	1	6.96 (0.00)	6.53 (0.00)	0.61 (0.00)
	$d_{ed4}$	0	1	7.59 (0.00)	4.69 (0.00)	0.74 (0.00)
Gaussian		0	20	21.11 (13.95)	3.97 (3.89)	0.69 (0.41)
	$d_{ed1}$	0	5	14.68 (7.81)	4.53 (4.46)	0.64 (0.46)
	$d_{ed2}$	0	5	14.81 (5.17)	2.81 (3.57)	0.83 (0.37)
	$d_{ed3}$	0	5	26.33 (16.86)	4.49 (4.46)	0.64 (0.46)
	$d_{ed4}$	0	5	28.61 (18.63)	4.06 (4.12)	0.68 (0.44)
Rat Quad		0	12	7.65 (1.51)	6.31 (2.93)	0.57 (0.30)
	$d_{ed1}$	0	3	6.96 (0.63)	7.65 (1.84)	0.44 (0.27)
	$d_{ed2}$	0	3	9.07 (2.40)	3.98 (4.18)	0.81 (0.31)
	$d_{ed3}$	0	3	6.75 (0.97)	7.40 (2.06)	0.47 (0.29)
	$d_{ed4}$	0	3	7.81 (0.73)	6.20 (3.06)	0.55 (0.35)
Poly		0	12	7.91 (0.99)	4.56 (2.50)	0.84 (0.19)
	$d_{ed1}$	0	3	7.59 (0.63)	5.72 (2.74)	0.77 (0.25)
	$d_{ed2}$	0	3	8.86 (1.10)	2.03 (0.60)	0.97 (0.03)
	$d_{ed3}$	0	3	8.23 (0.63)	5.62 (2.72)	0.78 (0.24)
	$d_{ed4}$	0	3	6.96 (0.63)	4.87 (2.36)	0.81 (0.19)

Table C.2: Static kernel assessment Recall dataset

Method	Distance	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
Pseudo		0	4	81.65 (14.90)	0.00 (0.00)	0.90 (0.05)
	$d_{ed1}$	0	1	95.57 (0.00)	0.00 (0.00)	0.89 (0.00)
	$d_{ed2}$	0	1	91.77 (0.00)	0.00 (0.00)	0.97 (0.00)
	$d_{ed3}$	0	1	75.95 (0.00)	0.00 (0.00)	0.89 (0.00)
	$d_{ed4}$	0	1	63.29 (0.00)	0.00 (0.00)	0.84 (0.00)
Similarity		0	4	3.16 (0.90)	5.08 (2.00)	0.81 (0.12)
	$d_{ed1}$	0	1	2.53 (0.00)	6.46 (0.00)	0.73 (0.00)
	$d_{ed2}$	0	1	4.43 (0.00)	2.19 (0.00)	0.99 (0.00)
	$d_{ed3}$	0	1	3.16 (0.00)	6.37 (0.00)	0.73 (0.00)
	$d_{ed4}$	0	1	2.53 (0.00)	5.30 (0.00)	0.79 (0.00)
Gaussian		0	20	11.93 (12.68)	4.38 (4.40)	0.69 (0.42)
	$d_{ed1}$	0	5	2.78 (0.72)	4.79 (4.99)	0.65 (0.47)

Kernel Evaluation Refer dataset



Table C.3 –

Method	Distance	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
Rat Quad	$d_{ed2}$	0	5	6.46 (2.07)	3.38 (4.17)	0.81 (0.40)
	$d_{ed3}$	0	5	15.57 (11.08)	4.77 (4.97)	0.65 (0.47)
	$d_{ed4}$	0	5	22.91 (18.07)	4.60 (4.81)	0.66 (0.46)
		0	12	3.32 (1.12)	6.72 (3.18)	0.61 (0.33)
	$d_{ed1}$	0	3	2.32 (0.73)	7.62 (2.64)	0.54 (0.36)
	$d_{ed2}$	0	3	4.22 (1.59)	4.85 (4.81)	0.76 (0.40)
	$d_{ed3}$	0	3	3.16 (0.63)	7.56 (2.69)	0.55 (0.36)
	$d_{ed4}$	0	3	3.59 (0.73)	6.86 (3.28)	0.58 (0.38)
Poly		0	12	3.22 (1.28)	4.94 (2.56)	0.86 (0.17)
	$d_{ed1}$	0	3	2.95 (0.37)	5.75 (3.03)	0.81 (0.22)
	$d_{ed2}$	0	3	4.85 (0.97)	2.96 (1.33)	0.97 (0.04)
	$d_{ed3}$	0	3	2.53 (1.10)	5.72 (3.02)	0.81 (0.22)
	$d_{ed4}$	0	3	2.53 (1.10)	5.33 (2.83)	0.83 (0.19)

Table C.3: Static kernel assessment Refer dataset

Method	Distance	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
Pseudo		0	4	76.27 (23.93)	0.00 (0.00)	0.90 (0.06)
	$d_{ed1}$	0	1	98.73 (0.00)	0.00 (0.00)	0.89 (0.00)
	$d_{ed2}$	0	1	93.67 (0.00)	0.00 (0.00)	0.98 (0.00)
	$d_{ed3}$	0	1	63.92 (0.00)	0.00 (0.00)	0.88 (0.00)
	$d_{ed4}$	0	1	48.73 (0.00)	0.00 (0.00)	0.84 (0.00)
Similarity		2	2	2.85 (0.45)	8.36 (4.14)	0.60 (0.26)
	$d_{ed1}$	0	1	0.00 (0.00)	11.20 (0.00)	0.41 (0.00)
	$d_{ed2}$	1	0	2.53 (0.00)	2.33 (0.00)	0.99 (0.00)
	$d_{ed3}$	0	1	0.00 (0.00)	10.92 (0.00)	0.46 (0.00)
	$d_{ed4}$	1	0	3.16 (0.00)	8.99 (0.00)	0.54 (0.00)
Gaussian		5	15	18.23 (20.58)	5.36 (5.32)	0.66 (0.43)
	$d_{ed1}$	2	3	0.38 (0.35)	6.01 (5.95)	0.60 (0.48)
	$d_{ed2}$	1	4	3.92 (2.43)	3.84 (4.97)	0.81 (0.41)
	$d_{ed3}$	1	4	19.75 (17.93)	5.94 (6.00)	0.61 (0.48)
	$d_{ed4}$	1	4	30.63 (26.85)	5.67 (5.88)	0.62 (0.48)
Rat Quad		2	10	1.65 (1.41)	9.54 (3.69)	0.46 (0.30)
	$d_{ed1}$	0	3	0.63 (0.00)	11.47 (0.92)	0.33 (0.21)
	$d_{ed2}$	1	2	1.90 (1.67)	5.53 (5.76)	0.76 (0.40)
	$d_{ed3}$	1	2	0.42 (0.37)	11.25 (1.11)	0.36 (0.23)
	$d_{ed4}$	0	3	2.53 (1.90)	9.90 (2.27)	0.41 (0.27)
Poly		6	6	1.90 (1.06)	6.01 (3.39)	0.84 (0.21)
	$d_{ed1}$	3	0	0.00 (0.00)	7.08 (3.92)	0.79 (0.26)
	$d_{ed2}$	0	3	2.32 (0.37)	3.21 (1.57)	0.97 (0.03)
	$d_{ed3}$	2	1	0.21 (0.37)	7.03 (3.92)	0.79 (0.26)

Kernel Evaluation Repeat dataset

Table C.4 –

Method	Distance	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
	$d_{ed4}$	1	2	1.27 (1.67)	6.71 (3.75)	0.80 (0.24)

Table C.4: Static kernel assessment Repeat dataset

Method	Distance	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
Pseudo		0	4	69.30 (16.77)	0.00 (0.00)	0.75 (0.20)
	$d_{ed1}$	0	1	81.01 (0.00)	0.00 (0.00)	0.85 (0.00)
	$d_{ed2}$	0	1	85.44 (0.00)	0.00 (0.00)	0.97 (0.00)
	$d_{ed3}$	0	1	60.76 (0.00)	0.00 (0.00)	0.63 (0.00)
	$d_{ed4}$	0	1	50.00 (0.00)	0.00 (0.00)	0.54 (0.00)
Similarity		0	4	15.35 (23.18)	4.19 (3.16)	0.68 (0.26)
	$d_{ed1}$	0	1	1.90 (0.00)	8.41 (0.00)	0.36 (0.00)
	$d_{ed2}$	0	1	6.33 (0.00)	1.83 (0.00)	0.99 (0.00)
	$d_{ed3}$	0	1	3.16 (0.00)	4.83 (0.00)	0.66 (0.00)
	$d_{ed4}$	0	1	50.00 (0.00)	1.70 (0.00)	0.69 (0.00)
Gaussian		0	20	21.46 (17.36)	3.53 (3.84)	0.71 (0.36)
	$d_{ed1}$	0	5	8.23 (4.84)	5.16 (4.72)	0.56 (0.44)
	$d_{ed2}$	0	5	10.00 (4.06)	3.17 (4.15)	0.81 (0.40)
	$d_{ed3}$	0	5	23.42 (17.30)	4.43 (4.49)	0.65 (0.41)
	$d_{ed4}$	0	5	44.18 (7.73)	1.38 (0.38)	0.81 (0.21)
Rat Quad		0	12	14.50 (18.87)	5.44 (3.79)	0.56 (0.29)
	$d_{ed1}$	0	3	4.01 (0.37)	8.96 (1.40)	0.28 (0.16)
	$d_{ed2}$	0	3	4.85 (2.22)	4.51 (4.78)	0.79 (0.35)
	$d_{ed3}$	0	3	3.80 (0.00)	6.55 (3.45)	0.49 (0.31)
	$d_{ed4}$	0	3	45.36 (6.97)	1.73 (0.07)	0.66 (0.08)
Poly		0	12	14.45 (19.16)	3.88 (2.65)	0.82 (0.17)
	$d_{ed1}$	0	3	3.80 (0.63)	6.30 (3.07)	0.76 (0.27)
	$d_{ed2}$	0	3	4.22 (0.37)	2.40 (0.89)	0.98 (0.02)
	$d_{ed3}$	0	3	3.59 (0.73)	5.14 (2.61)	0.81 (0.19)
	$d_{ed4}$	0	3	46.20 (1.10)	1.68 (0.09)	0.75 (0.11)

Table C.5: Static kernel assessment Test dataset

Method	Distance	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
Pseudo		0	4	72.31 (27.74)	0.00 (0.00)	0.80 (0.15)
	$d_{ed1}$	0	1	99.37 (0.00)	0.00 (0.00)	0.84 (0.00)
	$d_{ed2}$	0	1	93.04 (0.00)	0.00 (0.00)	0.98 (0.00)
	$d_{ed3}$	0	1	50.00 (0.00)	0.00 (0.00)	0.76 (0.00)
	$d_{ed4}$	0	1	46.84 (0.00)	0.00 (0.00)	0.63 (0.00)

Kernel Evaluation Therapy dataset

Table C.6 –

Method	Distance	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
Similarity		2	2	11.08 (11.19)	8.05 (5.26)	0.41 (0.40)
	$d_{ed1}$	1	0	0.00 (0.00)	12.53 (0.00)	0.12 (0.00)
	$d_{ed2}$	0	1	3.16 (0.00)	2.25 (0.00)	0.99 (0.00)
	$d_{ed3}$	1	0	0.00 (0.00)	12.48 (0.00)	0.16 (0.00)
	$d_{ed4}$	0	1	18.99 (0.00)	4.94 (0.00)	0.38 (0.00)
Gaussian		8	12	24.89 (22.47)	5.76 (4.97)	0.57 (0.41)
	$d_{ed1}$	5	0	0.00 (0.00)	7.99 (5.50)	0.41 (0.42)
	$d_{ed2}$	1	4	4.30 (2.74)	3.80 (5.01)	0.81 (0.41)
	$d_{ed3}$	2	3	20.63 (26.30)	7.43 (5.64)	0.47 (0.45)
	$d_{ed4}$	0	5	34.81 (21.89)	3.83 (3.30)	0.58 (0.39)
Rat Quad		7	5	10.13 (8.44)	9.24 (4.41)	0.33 (0.33)
	$d_{ed1}$	3	0	0.00 (0.00)	12.54 (0.03)	0.11 (0.03)
	$d_{ed2}$	1	2	2.11 (1.83)	5.49 (5.82)	0.76 (0.39)
	$d_{ed3}$	3	0	0.00 (0.00)	12.49 (0.07)	0.14 (0.05)
	$d_{ed4}$	0	3	14.77 (7.86)	6.45 (2.83)	0.31 (0.14)
Poly		6	6	8.86 (7.22)	5.60 (3.22)	0.80 (0.23)
	$d_{ed1}$	3	0	0.00 (0.00)	7.40 (3.98)	0.77 (0.28)
	$d_{ed2}$	0	3	2.32 (0.37)	3.10 (1.47)	0.97 (0.03)
	$d_{ed3}$	3	0	0.00 (0.00)	7.36 (3.98)	0.77 (0.28)
	$d_{ed4}$	0	3	15.40 (1.32)	4.53 (1.53)	0.70 (0.25)

Table C.6: Static kernel assessment Therapy dataset

Method	Distance	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
Pseudo		0	4	75.79 (24.99)	0.00 (0.00)	0.71 (0.35)
	$d_{ed1}$			99.37 (0.00)	0.00 (0.00)	0.89 (0.00)
	$d_{ed2}$			94.94 (0.00)	0.00 (0.00)	0.98 (0.00)
	$d_{ed3}$			58.86 (0.00)	0.00 (0.00)	0.75 (0.00)
	$d_{ed4}$			50.00 (0.00)	0.00 (0.00)	0.21 (0.00)
Similarity		2	2	26.58 (34.01)	6.84 (5.97)	0.60 (0.43)
	$d_{ed1}$	1	0	0.00 (0.00)	12.57 (0.00)	0.10 (0.00)
	$d_{ed2}$	0	1	2.53 (0.00)	2.23 (0.00)	0.99 (0.00)
	$d_{ed3}$	1	0	0.00 (0.00)	11.38 (0.00)	0.40 (0.00)
	$d_{ed4}$	0	1	50.63 (0.00)	1.16 (0.00)	0.92 (0.00)
Gaussian		8	12	29.85 (21.69)	4.95 (5.21)	0.67 (0.41)
	$d_{ed1}$	5	0	0.00 (0.00)	8.86 (5.22)	0.36 (0.41)
	$d_{ed2}$	1	4	3.16 (1.95)	3.80 (5.01)	0.81 (0.41)
	$d_{ed3}$	2	3	18.86 (20.71)	6.06 (5.99)	0.58 (0.47)
	$d_{ed4}$	0	5	49.62 (0.57)	1.11 (0.13)	0.94 (0.08)
Rat Quad		7	5	31.14 (26.12)	7.72 (5.46)	0.52 (0.39)
	$d_{ed1}$	3	0	0.00 (0.00)	12.57 (0.00)	0.09 (0.01)

Kernel Evaluation All data dataset

Table C.7 –

Method	Distance	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
Poly	$d_{ed2}$	1	2	1.69 (1.46)	5.47 (5.83)	0.77 (0.39)
	$d_{ed3}$	3	0	0.00 (0.00)	11.61 (0.86)	0.32 (0.20)
	$d_{ed4}$	0	3	50.21 (0.37)	1.22 (0.10)	0.89 (0.07)
		6	6	25.95 (26.35)	4.70 (3.72)	0.86 (0.19)
	$d_{ed1}$	3	0	0.00 (0.00)	7.42 (3.98)	0.77 (0.29)
	$d_{ed2}$	0	3	1.90 (0.63)	3.08 (1.46)	0.98 (0.03)
	$d_{ed3}$	3	0	0.00 (0.00)	7.13 (3.96)	0.79 (0.26)
	$d_{ed4}$	0	3	50.00 (0.00)	1.18 (0.03)	0.92 (0.03)

Table C.7: Static kernel assessment All data dataset

Method	Distance	PSD	NSD	%-ve Eig $\pm$ (std)	SpecR $\pm$ (std)	KTA $\pm$ (std)
Pseudo		0	4	61.39 (8.88)	0.00 (0.00)	0.72 (0.23)
	$d_{ed1}$	0	1	69.62 (0.00)	0.00 (0.00)	0.72 (0.00)
	$d_{ed2}$	0	1	67.09 (0.00)	0.00 (0.00)	0.97 (0.00)
	$d_{ed3}$	0	1	58.86 (0.00)	0.00 (0.00)	0.75 (0.00)
	$d_{ed4}$	0	1	50.00 (0.00)	0.00 (0.00)	0.42 (0.00)
Similarity		0	4	21.99 (20.08)	4.05 (3.25)	0.76 (0.29)
	$d_{ed1}$	0	1	5.06 (0.00)	8.35 (0.00)	0.34 (0.00)
	$d_{ed2}$	0	1	20.25 (0.00)	1.78 (0.00)	0.99 (0.00)
	$d_{ed3}$	0	1	12.03 (0.00)	4.75 (0.00)	0.78 (0.00)
	$d_{ed4}$	0	1	50.63 (0.00)	1.31 (0.00)	0.91 (0.00)
Gaussian		0	44	29.59 (16.54)	3.23 (3.46)	0.75 (0.36)
	$d_{ed1}$	0	11	17.61 (11.84)	4.71 (3.93)	0.58 (0.41)
	$d_{ed2}$	0	11	24.63 (12.30)	3.14 (3.63)	0.81 (0.36)
	$d_{ed3}$	0	11	27.56 (16.84)	3.86 (3.93)	0.71 (0.41)
	$d_{ed4}$	0	11	48.56 (4.00)	1.21 (0.28)	0.90 (0.16)
Rat Quad		0	56	21.55 (18.61)	5.23 (4.04)	0.55 (0.38)
	$d_{ed1}$	0	14	9.96 (10.85)	10.37 (3.77)	4.09 (0.38)
	$d_{ed2}$	0	14	14.00 (13.34)	6.12 (4.16)	0.81 (0.42)
	$d_{ed3}$	0	14	15.32 (16.36)	6.49 (4.03)	0.50 (0.41)
	$d_{ed4}$	0	14	43.81 (6.47)	1.80 (0.34)	0.75 (0.19)
Poly		0	12	19.15 (19.41)	3.68 (2.58)	0.86 (0.17)
	$d_{ed1}$	0	3	4.22 (0.37)	6.09 (2.78)	0.75 (0.28)
	$d_{ed2}$	0	3	14.98 (3.87)	2.32 (0.83)	0.98 (0.02)
	$d_{ed3}$	0	3	6.96 (2.53)	4.96 (2.49)	0.83 (0.19)
	$d_{ed4}$	0	3	50.42 (0.37)	1.35 (0.08)	0.89 (0.06)

Table C.8: Static kernel assessment (Common symbols) All data dataset

# References

- [1] FINDRISC (Finnish Diabetes Risk Score) - MDCalc. URL <https://www.mdcalc.com/findrisc-finnish-diabetes-risk-score>.
- [2] QDiabetes-2018 Risk Calculator. URL <https://qdiabetes.org/>.
- [3] J. Adler-Milstein and D. W. Bates. Paperless healthcare: Progress and challenges of an IT-enabled healthcare system. *Business Horizons*, 53(2):119–130, 2010. ISSN 00076813. doi: 10.1016/j.bushor.2009.10.004.
- [4] J. Adler-Milstein, C. M. DesRoches, M. F. Furukawa, C. Worzala, D. Charles, P. Kralovec, S. Stalley, and A. K. Jha. More than half of US hospitals have at least a basic EHR, but stage 2 criteria remain challenging for most. *Health Affairs*, 33(9):1664–1671, 2014. ISSN 15445208. doi: 10.1377/hlthaff.2014.0453.
- [5] R. Agrawal, C. Faloutsos, and A. Swami. Efficient similarity search in sequence databases. volume 5, pages 69–84. 1993. ISBN 9783540573012. doi: 10.1007/3-540-57301-1\_5. URL [http://link.springer.com/10.1007/3-540-57301-1\\_{\\_}5](http://link.springer.com/10.1007/3-540-57301-1_{_}5).
- [6] A. M. Alaa, T. Bolton, E. D. Angelantonio, J. H. Rudd, and M. van der Schaar. Cardiovascular disease risk prediction using automated machine learning: A prospective study of 423,604 UK Biobank participants. *PLoS ONE*, 14(5):1–17, 2019. ISSN 19326203. doi: 10.1371/journal.pone.0213653.
- [7] K. G. M. M. Alberti, P. Zimmet, and J. Shaw. International Diabetes Federation: A consensus on Type 2 diabetes prevention. *Diabetic Medicine*, 24(5):451–463, 2007. ISSN 07423071. doi: 10.1111/j.1464-5491.2007.02157.x.
- [8] M. Alghamdi, M. Al-Mallah, S. Keteyian, C. Brawner, J. Ehrman, and S. Sakr. Predicting diabetes mellitus using SMOTE and ensemble machine learning approach: The Henry Ford Exercise Testing (FIT) project. *PLoS ONE*, 12(7):1–15, 2017. ISSN 19326203. doi: 10.1371/journal.pone.0179805.
- [9] S. Ali and V. Fonseca. Overview of metformin: special focus on metformin extended release. *Expert opinion on pharmacotherapy*, 13(12):1797–805, aug 2012. ISSN 1744-7666. doi: 10.1517/14656566.2012.705829. URL <http://www.tandfonline.com/doi/full/10.1517/14656566.2012.705829><http://www.ncbi.nlm.nih.gov/pubmed/22775758>.
- [10] J. L. M. Amaral, A. J. Lopes, J. M. Jansen, A. C. D. Faria, and P. L. Melo. An improved method of early diagnosis of smoking-induced respiratory changes using machine learning algorithms. *Computer Methods and Programs in Biomedicine*, 112(3):441–454, 2013. ISSN 01692607. doi: 10.1016/j.cmpb.2013.08.004. URL <http://dx.doi.org/10.1016/j.cmpb.2013.08.004>.
- [11] S. E. Andrade, K. H. Kahler, F. Frech, and K. A. Chan. Methods for evaluation of medication adherence and persistence using automated databases. *Pharmacoepidemiology and Drug Safety*, 15(8):565–574, 2006. ISSN 10538569. doi: 10.1002/pds.1230.

- [12] H. Y. Ann, C. B. Yang, Y. H. Peng, and B. C. Liaw. Efficient algorithms for the block edit problems. *Information and Computation*, 208(3):221–229, 2010. ISSN 08905401. doi: 10.1016/j.ic.2009.12.001. URL <http://dx.doi.org/10.1016/j.ic.2009.12.001>.
- [13] A. K. Arslan, C. Colak, and M. E. Sarihan. Different medical data mining approaches based prediction of ischemic stroke. *Computer Methods and Programs in Biomedicine*, 130:87–92, 2016. ISSN 01692607. doi: 10.1016/j.cmpb.2016.03.022. URL <http://www.scopus.com/inward/record.url?eid=2-s2.0-84962288912&partnerID=tZOtx3y1>.
- [14] A. N. Arslan and O. Egecioglu. An efficient uniform-cost normalized edit distance algorithm. *String Processing and Information Retrieval Symposium and International Workshop on Groupware, SPIRE 1999 and CRIWG 1999*, pages 8–15, 1999. doi: 10.1109/SPIRE.1999.796572.
- [15] S. Aseervatham and Y. Bennani. Semi-structured document categorization with a semantic kernel. *Pattern Recognition*, 42(9):2067–2076, 2009. ISSN 00313203. doi: 10.1016/j.patcog.2008.10.024.
- [16] C. Auffray, R. Balling, I. Barroso, L. Bencze, M. Benson, and J. Bergeron. Making sense of big data in health research : Towards an EU action plan. *Genome Medicine*, pages 1–13, 2016. ISSN 1756-994X. doi: 10.1186/s13073-016-0323-y. URL <http://dx.doi.org/10.1186/s13073-016-0323-y>.
- [17] F. R. Bach, G. R. G. Lanckriet, and M. I. Jordan. Multiple kernel learning, conic duality, and the SMO algorithm. *Twentyfirst international conference on Machine learning ICML 04*, 69(1):6, 2004. ISSN 1581138285. doi: 10.1145/1015330.1015424. URL <http://portal.acm.org/citation.cfm?doid=1015330.1015424>.
- [18] J. Badger, E. LaRose, J. Mayer, F. Bashiri, D. Page, and P. Peissig. Machine learning for phenotyping opioid overdose events. *Journal of Biomedical Informatics*, 94(July 2018):103185, 2019. ISSN 15320464. doi: 10.1016/j.jbi.2019.103185. URL <https://doi.org/10.1016/j.jbi.2019.103185>.
- [19] F. Bagattini, I. Karlsson, J. Rebane, and P. Papapetrou. A classification framework for exploiting sparse multi-variate temporal features with application to adverse drug event detection in medical records. *BMC Medical Informatics and Decision Making*, 19(1):1–20, 2019. ISSN 14726947. doi: 10.1186/s12911-018-0717-4.
- [20] C. Bahlmann, B. Haasdonk, and H. Burkhardt. Online handwriting recognition with support vector machines - A kernel approach. *Proceedings - International Workshop on Frontiers in Handwriting Recognition, IWFHR*, pages 49–54, 2002. ISSN 15505235. doi: 10.1109/IWFHR.2002.1030883.
- [21] W. Bailer. Learning multiple sequence-based kernels for video concept detection. *Proceedings - 2012 IEEE International Symposium on Multimedia, ISM 2012*, pages 73–77, 2012. doi: 10.1109/ISM.2012.22.
- [22] K. Balhaf, M. A. Alsmirat, M. Al-Ayyoub, Y. Jararweh, and M. A. Shehab. Accelerating Levenshtein and Damerau edit distance algorithms using GPU with unified memory. *2017 8th International Conference on Information and Communication Systems, ICICS 2017*, pages 7–11, 2017. doi: 10.1109/IACS.2017.7921937.
- [23] L. Ballan, M. Bertini, A. Del Bimbo, and G. Serra. Video event classification using string kernels. *Multimedia Tools and Applications*, 48(1):69–87, 2010. ISSN 13807501. doi: 10.1007/s11042-009-0351-3.

- [24] Y. Bar-Dayana, H. Saed, M. Boaz, Y. Misch, T. Shahar, I. Husiascky, and O. Blumenfeld. Using electronic health records to save money. *Journal of the American Medical Informatics Association*, 20(E1):17–20, 2013. ISSN 10675027. doi: 10.1136/amiajnl-2012-001504.
- [25] S. R. Barber, M. J. Davies, K. Khunti, and L. J. Gray. Risk assessment tools for detecting those with pre-diabetes: A systematic review. *Diabetes Research and Clinical Practice*, 105(1):1–13, 2014. ISSN 18728227. doi: 10.1016/j.diabres.2014.03.007. URL <http://dx.doi.org/10.1016/j.diabres.2014.03.007>.
- [26] C. Barbieri, F. Mari, A. Stopper, E. Gatti, P. Escandell-Montero, J. M. Martínez-Martínez, and J. D. Martín-Guerrero. A new machine learning approach for predicting the response to anemia treatment in a large cohort of End Stage Renal Disease patients undergoing dialysis. *Computers in Biology and Medicine*, 61:56–61, 2015. ISSN 00104825. doi: 10.1016/j.combiomed.2015.03.019. URL <http://linkinghub.elsevier.com/retrieve/pii/S0010482515000979>.
- [27] G. V. Bard. Spelling-error tolerant, order-independent pass-phrases via the Damerau-Levenshtein string-edit distance metric. *Conferences in Research and Practice in Information Technology Series*, 68:117–124, 2007. ISSN 14451336.
- [28] I. M. Baytas, C. Xiao, X. Zhang, F. Wang, A. K. Jain, and J. Zhou. Patient subtyping via time-aware LSTM networks. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Part F1296:65–74, 2017. doi: 10.1145/3097983.3097997.
- [29] B. K. Beaulieu-Jones, P. Orzechowski, and J. H. Moore. Mapping patient trajectories using longitudinal extraction and deep learning in the MIMIC-III critical care database. *Pacific Symposium on Biocomputing*, 0(212669):123–132, 2018. ISSN 23356936. doi: 10.1142/9789813235533\_0012.
- [30] A. Bellet, M. Bernard, T. Murgue, and M. Sebban. Learning state machine-based string edit kernels. *Pattern Recognition*, 43(6):2330–2339, 2010. ISSN 00313203. doi: 10.1016/j.patcog.2009.12.008.
- [31] A. Bellet, M. Sebban, and A. Habrard. An experimental study on learning with good edit similarity functions. *Proceedings - International Conference on Tools with Artificial Intelligence, ICTAI*, pages 126–133, 2011. ISSN 10823409. doi: 10.1109/ICTAI.2011.27.
- [32] W. L. Bennett, N. M. Maruthur, S. Singh, J. B. Segal, L. M. Wilson, R. Chatterjee, S. S. Marinopoulos, M. A. Puhon, P. Ranasinghe, L. Block, W. K. Nicholson, S. Hutflless, E. B. Bass, and S. Bolen. Comparative effectiveness and safety of medications for type 2 diabetes: An update including new drugs and 2-drug combinations, may 2011. ISSN 00034819. URL <http://annals.org/article.aspx?doi=10.7326/0003-4819-154-9-201105030-00336>.
- [33] M. Bernard, J.-c. Janodet, and M. Sebban. A Discriminative Model of Stochastic Edit Distance in the Form of a Conditional Transducer. *Network*, pages 240–252, 2006. ISSN 03029743.
- [34] D. Berndt and J. Clifford. Using dynamic time warping to find patterns in time series. *Workshop on Knowledge Knowledge Discovery in Databases*, 398:359–370, 1994. URL <http://www.aai.org/Papers/Workshops/1994/WS-94-03/WS94-03-031.pdf>.
- [35] M. Bilenko and R. J. Mooney. Learning to Combine Trained Distance Metrics for Duplicate Detection in Databases. *Artificial Intelligence*, (February):19, 2002.
- [36] L. Blonde, G. E. Dailey, S. A. Jabbour, C. A. Reasner, and D. J. Mills. Gastrointestinal tolerability of extended-release metformin tablets compared to immediate-release metformin tablets: results of a retrospective cohort study. *Current Medical Research and Opinion*, 20

- (4):565–572, apr 2004. ISSN 0300-7995. doi: 10.1185/030079904125003278. URL <http://www.tandfonline.com/doi/full/10.1185/030079904125003278>.
- [37] D. Blumenthal. Stimulating the Adoption of Health Information Technology. *New England Journal of Medicine*, 360(15):1477–1479, 2009. ISSN 0028-4793. doi: 10.1056/nejmp0901592. URL <https://doi.org/10.1056/NEJMp0901592>.
- [38] S. J. Boccuzzi, J. Wogen, J. Fox, J. C. Sung, A. B. Shah, and J. Kim. Utilization of oral hypoglycemic agents in a drug-insured U.S. population. *Diabetes Care*, 24(8):1411–1415, aug 2001. ISSN 01495992. doi: 10.2337/diacare.24.8.1411. URL <http://www.ncbi.nlm.nih.gov/pubmed/11473078>.
- [39] T. Botsis, G. Hartvigsen, F. Chen, and C. Weng. Secondary Use of EHR: Data Quality Issues and Informatics Opportunities. *AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science*, 2010:1–5, 2010. ISSN 2153-4063. URL <http://www.ncbi.nlm.nih.gov/pubmed/21347133>{%}0A<http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC3041534>.
- [40] A. A. Boxwala, J. Kim, J. M. Grillo, and L. Ohno-Machado. Using statistical and machine learning to help institutions detect suspicious access to electronic health records. *Journal of the American Medical Informatics Association*, 18(4):498–505, 2011. ISSN 10675027. doi: 10.1136/amiajnl-2011-000217.
- [41] S. Boyle. United Kingdom (England) Health system review. *Health systems in transition*, 13, 2011.
- [42] T. Bozkaya, N. Yazdani, and M. Özsoyoglu. Matching and Indexing Sequences of Different Lengths. *Cikm*, pages 128–135, 1997. doi: 10.1145/266714.266880. URL <http://portal.acm.org/citation.cfm?doid=266714.266880>.
- [43] H. Bunke and K. Riesen. Towards the unification of structural and statistical pattern recognition. *Pattern Recognition Letters*, 33(7):811–825, 2012. ISSN 01678655. doi: 10.1016/j.patrec.2011.04.017. URL <http://dx.doi.org/10.1016/j.patrec.2011.04.017>.
- [44] D. Bzdok and A. Meyer-Lindenberg. Machine Learning for Precision Psychiatry: Opportunities and Challenges. *Biological Psychiatry: Cognitive Neuroscience and Neuroimaging*, 3(3):223–230, 2018. ISSN 24519030. doi: 10.1016/j.bpsc.2017.11.007. URL <https://doi.org/10.1016/j.bpsc.2017.11.007>.
- [45] J. Cai, Y. Wang, O. Baser, L. Xie, and W. Chow. Comparative persistence and adherence with newer anti-hyperglycemic agents to treat patients with type 2 diabetes in the United States. *Journal of medical economics*, 6998(August):1–12, 2016. ISSN 1941-837X. doi: 10.1080/13696998.2016.1208208. URL <http://www.ncbi.nlm.nih.gov/pubmed/27356271>.
- [46] C. Campbell. Kernel methods: A survey of current techniques. *Neurocomputing*, 42:63–84, 2002. ISSN 09252312. doi: 10.1016/S0925-2312(01)00643-9.
- [47] J. M. Campbell, M. D. Stephenson, de Court, I. Chapman, S. M. Bellman, and E. Aromataris. Metformin and Alzheimer’s disease, dementia and cognitive impairment: a systematic review protocol. *JBI.Database.System.Rev.Implement.Rep.*, 15(8):2055–2059, 2017. ISSN 2202-4433. doi: 10.11124/JBISRIR-2017-003380.
- [48] J. A. Casey, B. S. Schwartz, W. F. Stewart, and N. E. Adler. Electronic Health Records and Population Health Research. *Frontiers in Public Health Services and Systems Research*, 5(5):15–22, 2016. doi: 10.13023/FPHSSR.0505.03. URL <http://uknowledge.uky.edu/frontiersinphssr>.



- [49] S. Chakraborty, R. Tomsett, R. Raghavendra, D. Harborne, M. Alzantot, F. Cerutti, M. Srivastava, A. Preece, S. Julier, R. M. Rao, T. D. Kelley, D. Braines, M. Sensoy, C. J. Willis, and P. Gurram. Interpretability of deep learning models: A survey of results. *2017 IEEE SmartWorld Ubiquitous Intelligence and Computing, Advanced and Trusted Computed, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People and Smart City Innovation, SmartWorld/SCALCOM/UIC/ATC/CBDCOM/IOP/SCI 2017 -*, pages 1–6, 2018. doi: 10.1109/UIC-ATC.2017.8397411.
- [50] K.-p. Chan and A. W.-c. Fu. Efficient Time Series Matching by Wavelets. *Proceedings 15th International Conference on Data Engineering (Cat. No.99CB36337)*, 0:126–133, 1999. doi: 10.1109/ICDE.1999.754915.
- [51] C.-c. Chang and C.-j. Lin. LIBSVM: A Library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2:1–39, 2011. ISSN 21576904. doi: 10.1145/1961189.1961199.
- [52] D. Charles, M. Gabriel, T. Searcy, N. Carolina, and S. Carolina. Adoption of Electronic Health Record Systems Among U.S. Non-Federal Acute Care Hospitals : 2008 -2014. 4(23):2008–2014, 2015. doi: ONCDDataBrief35. URL <https://www.healthit.gov/sites/default/files/data-brief/2014HospitalAdoptionDataBrief.pdf>.
- [53] W. Che, J. Jiang, Z. Su, Y. Pan, and T. Liu. Improved-edit-distance kernel for Chinese relation extraction. *Proceedings of the 2nd International Joint Conference on Natural Language Processing (IJCNLP'05)*, pages 134–139, 2005.
- [54] L. CHEN and R. NG. On The Marriage of Lp-norms and Edit Distance. *Proceedings 2004 VLDB Conference*, pages 792–803, 2004. doi: 10.1016/B978-012088469-8/50070-X. URL <http://linkinghub.elsevier.com/retrieve/pii/B978012088469850070X>.
- [55] L. Chen, M. T. Özsu, and V. Oria. Robust and fast similarity search for moving object trajectories. *Proceedings of the 2005 ACM SIGMOD international conference on Management of data - SIGMOD '05*, page 491, 2005. ISSN 07308078. doi: 10.1145/1066157.1066213. URL <http://portal.acm.org/citation.cfm?doid=1066157.1066213>.
- [56] E. Choi, A. Schuetz, W. F. Stewart, and J. Sun. Using recurrent neural network models for early detection of heart failure onset. *Journal of the American Medical Informatics Association*, 24(2):361–370, 2017. ISSN 1527974X. doi: 10.1093/jamia/ocw112.
- [57] F. Cismondi, A. S. Fialho, S. M. Vieira, S. R. Reti, J. M. C. Sousa, and S. N. Finkelstein. Missing data in medical databases: Impute, delete or classify? *Artificial Intelligence in Medicine*, 58(1): 63–72, 2013. ISSN 09333657. doi: 10.1016/j.artmed.2013.01.003. URL <http://dx.doi.org/10.1016/j.artmed.2013.01.003>.
- [58] G. L. Colombo, E. Agabiti-Rosei, A. Margonato, C. Mencacci, C. M. Montecucco, R. Trevisan, and A. L. Catapano. Impact of substitution among generic drugs on persistence and adherence: A retrospective claims data study from 2 Local Healthcare Units in the Lombardy Region of Italy. *Atherosclerosis Supplements*, 21:1–8, 2016. ISSN 18785050. doi: 10.1016/j.atherosclerosissup.2016.02.001. URL <http://dx.doi.org/10.1016/j.atherosclerosissup.2016.02.001>.
- [59] C. Combi and G. Pozzi. Clinical Information Systems and Artificial Intelligence: Recent Research Trends. *Yearbook of medical informatics*, 28(1):83–94, 2019. ISSN 23640502. doi: 10.1055/s-0039-1677915.

- [60] P. Coorevits, M. Sundgren, G. O. Klein, A. Bahr, B. Claerhout, C. Daniel, M. Dugas, D. Dupont, A. Schmidt, P. Singleton, G. De Moor, and D. Kalra. Electronic health records: New opportunities for clinical research. *Journal of Internal Medicine*, 274(6):547–560, 2013. ISSN 09546820. doi: 10.1111/joim.12119.
- [61] G. Cormode and S. Muthukrishnan. The string edit distance matching problem with moves. *ACM Transactions on Algorithms*, 3(1):1, 2007. ISSN 15496325. doi: 10.1145/1186810.1186812. URL <http://portal.acm.org/citation.cfm?doid=1186810.1186812>.
- [62] C. Cortes, P. Haffner, and M. Mohri. Rational Kernels : Theory and Algorithms. *Journal of Machine Learning Research*, 5:1035–1062, 2004. ISSN 15324435. URL <http://portal.acm.org/citation.cfm?id=1016793>.
- [63] C. Cortes, M. Mohri, and a. Rostamizadeh. Learning non-linear combinations of kernels. *Advances in Neural Information ...*, pages 1–9, 2009. URL <http://www.cs.nyu.edu/{~}mohri/pub/nlk.pdf>.
- [64] N. Cristianini, J. Kandola, A. Elisseeff, and J. Shawe-Taylor. On kernel-target alignment. *Advances in Neural Information Processing Systems 14*, pages 367—373, 2002. doi: 10.1.1.23.6757.
- [65] C. J. Currie, C. D. Poole, S. Jenkins-Jones, E. A. M. Gale, J. A. Johnson, and C. L. Morgan. Mortality after incident cancer in people with and without type 2 diabetes: Impact of metformin on survival. *Diabetes Care*, 35(2):299–304, 2012. ISSN 01495992. doi: 10.2337/dc11-1313.
- [66] M. Cuturi. Fast Global Alignment Kernels. *Review Literature And Arts Of The Americas*, pages 929–936, 2011. URL <http://www.iip.ist.i.kyoto-u.ac.jp/member/cuturi/Papers/cuturi11fast.pdf>.
- [67] M. Cuturi, J.-P. Vert, O. Birkenes, and T. Matsui. A kernel for time series based on global alignments. *Cs/0610033*, pages 1–9, 2006. doi: doi:10.1109/ICASSP.2007.366260. URL <http://arxiv.org/abs/cs/0610033><http://arxiv.org/pdf/cs/0610033v1>.
- [68] W. Dai, T. S. Brisimi, W. G. Adams, T. Mela, V. Saligrama, and I. C. Paschalidis. Prediction of hospitalization due to heart diseases by supervised learning methods. *International Journal of Medical Informatics*, 84(3):189–197, 2015. ISSN 13865056. doi: 10.1016/j.ijmedinf.2014.10.002. URL <http://linkinghub.elsevier.com/retrieve/pii/S1386505614001907>.
- [69] M. R. Daliri and V. Torre. Shape recognition and retrieval using string of symbols. *Proceedings - 5th International Conference on Machine Learning and Applications, ICMLA 2006*, pages 101–106, 2006. doi: 10.1109/ICMLA.2006.48.
- [70] M. R. Daliri and V. Torre. Shape recognition based on Kernel-edit distance. *Computer Vision and Image Understanding*, 114(10):1097–1103, 2010. ISSN 10773142. doi: 10.1016/j.cviu.2010.07.002. URL <http://dx.doi.org/10.1016/j.cviu.2010.07.002>.
- [71] F. J. Damerau. A technique for computer detection and correction of spelling errors. *Communications of the ACM*, 7(3):171–176, 1964. ISSN 00010782. doi: 10.1145/363958.363994. URL <http://portal.acm.org/citation.cfm?doid=363958.363994>.
- [72] I. Danciu, J. D. Cowan, M. Basford, X. Wang, A. Saip, S. Osgood, J. Shirey-Rice, J. Kirby, and P. a. Harris. Secondary use of clinical data: The Vanderbilt approach. *Journal of Biomedical Informatics*, 52:28–35, 2014. ISSN 15320464. doi: 10.1016/j.jbi.2014.02.003. URL <http://dx.doi.org/10.1016/j.jbi.2014.02.003>.

- [73] H. R. Darabi, D. Tsinis, K. Zecchini, W. F. Whitcomb, and A. Liss. Forecasting mortality risk for patients admitted to intensive care units using machine learning. *Procedia Computer Science*, 140:306–313, 2018. ISSN 18770509. doi: 10.1016/j.procs.2018.10.313. URL <https://doi.org/10.1016/j.procs.2018.10.313>.
- [74] I. M. De Diego, J. M. Moguerza, and A. Muñoz. Combining kernel information for support vector classification. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3077:102–111, 2004. ISSN 16113349. doi: 10.1007/978-3-540-25966-4\_10.
- [75] S. De Lusignan. Unleashing the power of e-Health requires the development of an evidence base for interventions that improve care LEADERS: UNLEASHING THE POWER OF E-HEALTH REQUIRES THE RIGHT EVIDENCE BASE. *Journal of Innovation in Health Informatics*, 22(1), 2015.
- [76] S. de Lusignan and C. van Weel. The use of routinely collected computer data for research in primary care: Opportunities and challenges. *Family Practice*, 23(2):253–263, 2006. ISSN 02632136. doi: 10.1093/fampra/cmi106.
- [77] S. De Lusignan, K. Khunti, J. Belsey, A. Hattersley, J. Van Vlymen, H. Gallagher, C. Millett, N. J. Hague, C. Tomson, K. Harris, and A. Majeed. A method of identifying and correcting miscoding, misclassification and misdiagnosis in diabetes: A pilot and validation study of routinely collected data. *Diabetic Medicine*, 27(2):203–209, 2010. ISSN 14645491. doi: 10.1111/j.1464-5491.2009.02917.x.
- [78] S. C. Denaxas and K. I. Morley. Big biomedical data and cardiovascular disease research: Opportunities and challenges, 2015. ISSN 20581742. URL <https://academic.oup.com/ehjqcco/article/1/1/9/1860292>.
- [79] C. M. DesRoches, E. G. Campbell, S. R. Rao, K. Donelan, T. G. Ferris, A. Jha, R. Kaushal, D. E. Levy, S. Rosenbaum, A. E. Shields, and D. Blumenthal. Electronic Health Records in Ambulatory Care — A National Survey of Physicians. *New England Journal of Medicine*, 359(1):50–60, 2008. ISSN 0028-4793. doi: 10.1056/nejmsa0802005.
- [80] Diabetes UK. Facts and Figures — Diabetes UK, 2017. URL <https://www.diabetes.org.uk/professionals/position-statements-reports/statisticshttps://www.diabetes.org.uk/Professionals/Position-statements-reports/Statistics>.
- [81] S. G. Díez, F. Foussy, M. Shimboz, and M. Saerens. Normalized sum-over-paths edit distances. *Proceedings - International Conference on Pattern Recognition*, pages 1044–1047, 2010. ISSN 10514651. doi: 10.1109/ICPR.2010.261.
- [82] H. Ding, G. Trajcevski, P. Scheuermann, X. Wang, and E. Keogh. Querying and mining of time series data: experimental comparison of representations and distance measures. *Proceedings of the VLDB Endowment*, 1(2):1542–1552, 2008. ISSN 2150-8097. doi: 10.1145/1454159.1454226. URL <http://dl.acm.org/citation.cfm?id=1454159.1454226%5Cnfile:///Users/bwilcox6/Dropbox/Thesis/Mekentosj/Library.papers3/Files/08/08A2DC46-9492-4520-BB79-9AED63E1370D.pdf%5Cnpapers3://publication/uuid/C5531CD9-BAC2-42E8-89C9-413B86FB7C35>.
- [83] I. D. Dinov. Methodological challenges and analytic opportunities for modeling and interpreting Big Healthcare Data. *GigaScience*, 5(1):1–15, 2016. ISSN 2047217X. doi: 10.1186/s13742-016-0117-6.

- [84] W. Y. Donglin, W. Peng, L. Ying, W. Chunhua, and Zeng. Learning Optimal Individualized Treatment Rules from Electronic Health Record Data. pages 65–71, 2017. doi: 10.1109/ICHI.2016.13.Learning.
- [85] M. Donini and F. Aioli. Learning deep kernels in the space of dot product polynomials. *Machine Learning*, 106(9):1245–1269, 2017. ISSN 1573-0565. doi: 10.1007/s10994-016-5590-8.
- [86] S. El-Sappagh and A. Farman. DDO: a diabetes mellitus diagnosis ontology. *Applied Informatics*, 3(5), 2006. ISSN 2196-0089. doi: 10.1186/s40535-016-0021-2.
- [87] N. Emanet, H. R. Öz, N. Bayram, and D. Delen. A comparative analysis of machine learning methods for classification type decision problems in healthcare. *Decision Analytics*, 1(1):6, 2014. ISSN 2193-8636. doi: 10.1186/2193-8636-1-6. URL <http://www.decisionanalyticsjournal.com/1/1/6>.
- [88] J. M. M. Evans, L. A. Donnelly, A. M. Emslie-Smith, D. R. Alessi, and A. D. Morris. Metformin and reduced risk of cancer in diabetic patients. *British Medical Journal*, 330(7503):1304–1305, jun 2005. ISSN 1756-1833. doi: 10.1136/bmj.38415.708634.F7. URL <http://www.ncbi.nlm.nih.gov/pubmed/15849206><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC558205><http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?cmd=Retrieve{%&}db=PubMed{%&}dopt=Citation{%&}list{-}uids=15849206{%}%}5Cn><http://www.pubmedcentral.nih.gov/artic>.
- [89] K. P. Exarchos, Y. Goletsis, and D. I. Fotiadis. Multiparametric decision support system for the prediction of oral cancer reoccurrence. *IEEE Transactions on Information Technology in Biomedicine*, 16(6):1127–1134, 2012. ISSN 10897771. doi: 10.1109/TITB.2011.2165076.
- [90] R. Farivar, H. Kharbanda, S. Venkataraman, and R. H. Campbell. An algorithm for fast edit distance computation on GPUs. *2012 Innovative Parallel Computing, InPar 2012*, pages 0–8, 2012. doi: 10.1109/InPar.2012.6339593.
- [91] M. D. Feher, M. Al-Mrayat, J. Brake, and K. S. Leong. Tolerability of prolonged-release metformin (Glucophage SR) in individuals intolerant to standard metformin - Results from four UK centres. *British Journal of Diabetes and Vascular Disease*, 7(5):225–228, 2007. ISSN 14746514. doi: 10.1177/14746514070070050501.
- [92] K. Filali and J. Bilmes. A Dynamic Bayesian Framework to Model Context and Memory in Edit Distance Learning : An Application to Pronunciation Classification. (June):338–345, 2005.
- [93] H. Florez, J. Luo, S. Castillo-Florez, G. Mitsi, J. Hanna, L. Tamariz, A. Palacio, S. Nagendran, and M. Hagan. Impact of Metformin-Induced Gastrointestinal Symptoms on Quality of Life and Adherence in Patients with Type 2 Diabetes. *Postgraduate Medicine*, 122(2):112–120, mar 2010. ISSN 0032-5481. doi: 10.3810/pgm.2010.03.2128. URL <http://www.tandfonline.com/doi/full/10.3810/pgm.2010.03.2128>.
- [94] Y. Fong, S. Datta, I. S. Georgiev, P. D. Kwong, and G. D. Tomaras. Kernel-based logistic regression model for protein sequence without vectorialization. *Biostatistics*, 16(3):480–492, 2015. ISSN 1465-4644. doi: 10.1093/biostatistics/kxu056. URL <http://biostatistics.oxfordjournals.org/content/16/3/480.abstract>.
- [95] K. Fujioka, R. L. Brazg, I. Raz, S. Bruce, S. Joyal, R. Swanink, and M. Pans. Efficacy, dose-response relationship and safety of once-daily extended-release metformin (Glucophage XR) in type 2 diabetic patients with inadequate glycaemic control despite prior treatment with diet and exercise: Results from two double-blind, placebo. *Diabetes, Obesity and Metabolism*, 7(1):28–39, jan 2005. ISSN 14628902. doi: 10.1111/j.1463-1326.2004.00369.x. URL <http://doi.wiley.com/10.1111/j.1463-1326.2004.00369.x>.

- [96] I. R. Galatzer-Levy, K.-I. Karstoft, A. Statnikov, and A. Y. Shalev. Quantitative forecasting of PTSD from early trauma responses: A Machine Learning application. *Journal of Psychiatric Research*, 59:68–76, 2014. ISSN 00223956. doi: 10.1016/j.jpsychires.2014.08.017. URL <http://linkinghub.elsevier.com/retrieve/pii/S002239561400260X>.
- [97] A. J. Garber, T. G. Duncan, A. M. Goodman, D. J. Mills, and J. L. Rohlf. Efficacy of metformin in type II diabetes: results of a double-blind, placebo-controlled, dose-response trial. *Am J Med*, 103(6):491–497, 1997. ISSN 00029343. doi: 10.1016/S0002-9343(97)00254-4. URL <http://www.ncbi.nlm.nih.gov/pubmed/9428832>.
- [98] J. M. Gesulga, A. Berjame, K. S. Moquiala, and A. Galido. Barriers to Electronic Health Record System Implementation and Information Systems Resources: A Structured Review. In *Procedia Computer Science*, 2017. doi: 10.1016/j.procs.2017.12.188.
- [99] A. Gkoulalas-Divanis, G. Loukides, and J. Sun. Toward smarter healthcare: Anonymizing medical data to support research studies. *IBM Journal of Research & Development*, 58(1):1–11, 2014. ISSN 0018-8646. doi: 10.1147/JRD.2013.2288173.
- [100] B. A. Goldstein, A. M. Navar, M. J. Pencina, and J. P. A. Ioannidis. Opportunities and challenges in developing risk prediction models with electronic health records data: a systematic review. *Journal of the American Medical Informatics Association*, 24(1):198–208, 2017. ISSN 1067-5027. doi: 10.1093/jamia/ocw042. URL <https://academic.oup.com/jamia/article-lookup/doi/10.1093/jamia/ocw042>.
- [101] M. Gönen and E. Alpaydm. Multiple Kernel Learning Algorithms. *Journal of Machine Learning Research*, 12:2211–2268, 2011. ISSN 15324435. URL [files/1477/JMLR-Gonen-Alpaydin-2011-Multiple-\\_-Kernel-\\_-Learning-\\_-Algorithms.pdf](files/1477/JMLR-Gonen-Alpaydin-2011-Multiple-_-Kernel-_-Learning-_-Algorithms.pdf).
- [102] J. Gong, L. A. Robbins, A. Lugea, R. T. Waldron, C. Y. Jeon, and S. J. Pandol. Diabetes, pancreatic cancer, and metformin therapy. *Frontiers in Physiology*, 5(OCT):1–8, 2014. ISSN 1664042X. doi: 10.3389/fphys.2014.00426.
- [103] J. Gong, G. E. Simon, and S. L. Id. Machine learning discovery of longitudinal patterns of depression and suicidal ideation. *PLOS ONE*, pages 1–15, 2019.
- [104] T. Graepel, R. Herbrich, P. Bollmann-Sdorra, and K. Obermayer. Classification on Pairwise Proximity Data. *Advances in Neural Information Processing Systems 11*, 11:438–444, 1999. URL <http://books.google.com/books?hl=en&lr=&id=bMuzXPzlkG0C&oi=fnd&pg=PA438&dq=Classification+on+Pairwise+Proximity+Data&ots=MvnhyCBIPi&sig=VlrAY0hS9e7RXEmLBFRvBh4r74>.
- [105] P. J. Grant and F. Cosentino. The 2019 ESC Guidelines on diabetes, pre-diabetes, and cardiovascular diseases developed in collaboration with the EASD. *European Heart Journal*, 40(39):3215–3217, 2019. ISSN 15229645. doi: 10.1093/eurheartj/ehz687.
- [106] L. Gravano, P. G. Ipeirotis, H. V. Jagadish, N. Koudas, S. Muthukrishnan, L. Pietarinen, and D. Srivastava. Using q-grams in a DBMS for Approximate String Processing. *IEEE Data Eng. Bull.*, 24(4):28–34, 2001. doi: 10.1.1.14.6009.
- [107] J. P. Grégoire, C. Sirois, G. Blanc, P. Poirier, and J. Moisan. Persistence patterns with oral antidiabetes drug treatment in newly treated patients - A population-based study. *Value in Health*, 13(6):820–828, 2010. ISSN 15244733. doi: 10.1111/j.1524-4733.2010.00761.x. URL <http://dx.doi.org/10.1111/j.1524-4733.2010.00761.x>.

- [108] F. Grisoni, C. S. Neuhaus, M. Hishinuma, G. Gabernet, J. A. Hiss, M. Kotera, and G. Schneider. De novo design of anticancer peptides by ensemble artificial neural networks. *Journal of Molecular Modeling* 2019 25:5, 25(5):1–10, apr 2019. ISSN 0948-5023. doi: 10.1007/S00894-019-4007-6. URL <https://link.springer.com/article/10.1007/s00894-019-4007-6>.
- [109] S. Gudmundsson, T. P. Runarsson, and S. Sigurdsson. Support vector machines and dynamic time warping for time series. *2008 IEEE International Joint Conference on Neural Networks*, (x):2772–2776, 2008. ISSN 1098-7576. doi: 10.1109/IJCNN.2008.4634188. URL <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4634188>.
- [110] E. Gultepe, J. P. Green, H. Nguyen, J. Adams, T. Albertson, and I. Tagkopoulos. From vital signs to clinical outcomes for patients with sepsis: A machine learning basis for a clinical decision support system. *Journal of the American Medical Informatics Association*, 21(2):315–325, 2014. ISSN 10675027. doi: 10.1136/amiajnl-2013-001815.
- [111] B. Haasdonk. Feature space interpretation of SVMs with indefinite kernels. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(4):482–492, 2005. ISSN 01628828. doi: 10.1109/TPAMI.2005.78.
- [112] M. Hashir and R. Sawhney. Towards unstructured mortality prediction with free-text clinical notes. *Journal of Biomedical Informatics*, 108(April):103489, 2020. ISSN 15320464. doi: 10.1016/j.jbi.2020.103489. URL <https://doi.org/10.1016/j.jbi.2020.103489>.
- [113] D. Haussler. Convolution Kernels on Discrete Structures. *Journal of Bioenergetics and Biomembranes*, 43(1):1–2, 1999. ISSN 0145479X. doi: 10.1007/s10863-011-9338-7.
- [114] T. Heart, O. Ben-Assuli, and I. Shabtai. A review of PHR, EMR and EHR integration: A more personalized healthcare and public health policy. *Health Policy and Technology*, 6(1):20–25, 2017. ISSN 22118845. doi: 10.1016/j.hlpt.2016.08.002.
- [115] W. R. Hersh, M. G. Weiner, P. J. Embi, J. R. Logan, P. R. Payne, E. V. Bernstam, H. P. Lehmann, G. Hripcsak, T. H. Hartzog, J. J. Cimino, and J. H. Saltz. Recommendations for the Use of Operational Electronic Health Record Data in Comparative Effectiveness Research. *Medical Care*, 51(8 SUPPL.3), 2013. ISSN 00257079. doi: 10.1097/MLR.0b013e31829b1dbd.
- [116] W. R. Hersh, M. G. Weiner, P. J. Embi, J. R. Logan, P. R. Payne, E. V. Bernstam, H. P. Lehmann, G. Hripcsak, T. H. Hartzog, J. J. Cimino, and J. H. Saltz. Caveats for the Use of Operational Electronic Health Record Data in Comparative Effectiveness Research. *Medical care*, 51(August):S30–S37, 2014. ISSN 1537-1948. doi: 10.1097/MLR.0b013e31829b1dbd.Caveats.
- [117] R. Hillestad, J. Bigelow, A. Bower, F. Girosi, R. Meili, R. Scoville, and R. Taylor. Can electronic medical record systems transform health care? Potential health benefits, savings, and costs. *Health Affairs*, 24(5):1103–1117, 2005. ISSN 02782715. doi: 10.1377/hlthaff.24.5.1103.
- [118] J. Hippisley-Cox and C. Coupland. Development and validation of QDiabetes-2018 risk prediction algorithm to estimate future risk of type 2 diabetes: cohort study. *BMJ (Clinical research ed.)*, 359:j5019, nov 2017. ISSN 17561833. doi: 10.1136/bmj.j5019.
- [119] J. C. Ho, J. Ghosh, S. R. Steinhubl, W. F. Stewart, J. C. Denny, B. A. Malin, and J. Sun. Limestone: High-throughput candidate phenotype generation via tensor factorization. *Journal of Biomedical Informatics*, 52:199–211, 2014. ISSN 15320464. doi: 10.1016/j.jbi.2014.07.001. URL <http://dx.doi.org/10.1016/j.jbi.2014.07.001>.
- [120] F. Hosseinkhah, H. Ashktorab, R. Veen, and M. M. Owrang O. Challenges in Data Mining on Medical Databases. *Database Technologies: Concepts, Methodologies, Tools, and Applications*,

- pages 1393–1404, 2009. doi: 10.4018/978-1-60566-058-5.ch083. URL <http://services.igi-global.com/resolvedoi/resolve.aspx?doi=10.4018/978-1-60566-058-5.ch083>.
- [121] H. C. S. Howlett and C. J. Bailey. A risk-benefit assessment of metformin in type 2 diabetes mellitus. *Drug Safety*, 20(6):489–503, jun 1999. ISSN 01145916. doi: 10.2165/00002018-199920060-00003. URL <http://www.ncbi.nlm.nih.gov/pubmed/10392666><http://web.b.ebscohost.com/ergo.glam.ac.uk/ehost/pdfviewer/pdfviewer?sid=278b5da6-99da-4af4-a766-f2bd673d8ea5@sessionmgr114&vid=1&hid=124>.
- [122] R. S. Hundal and S. E. Inzucchi. Metformin: New Understandings, New Uses, 2003. ISSN 00126667. URL <http://www.ncbi.nlm.nih.gov/pubmed/12930161>.
- [123] A. M. Hung, C. L. Roumie, R. A. Greevy, X. Liu, C. G. Grijalva, H. J. Murff, T. A. Ikizler, and M. R. Griffin. Comparative effectiveness of incident oral antidiabetic drugs on kidney function. *Kidney International*, 81(7):698–706, 2012. ISSN 00852538. doi: 10.1038/ki.2011.444. URL <http://linkinghub.elsevier.com/retrieve/pii/S0085253815553660>.
- [124] T. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. *Advances in neural information processing systems*, pages 487–493, 1999. ISSN 10495258. doi: 10.1038/217994a0.
- [125] S. Jabbour and B. Ziring. Advantages of extended-release metformin in patients with type 2 diabetes mellitus. *Postgraduate Medicine*, 123(1):15–23, jan 2011. ISSN 00325481. doi: 10.3810/pgm.2011.01.2241. URL <http://www.tandfonline.com/doi/full/10.3810/pgm.2011.01.2241>.
- [126] P. B. Jensen, L. J. Jensen, and S. Brunak. Mining electronic health records: Towards better research applications and clinical care. *Nature Reviews Genetics*, 13(6):395–405, 2012. ISSN 14710056. doi: 10.1038/nrg3208. URL <http://dx.doi.org/10.1038/nrg3208>.
- [127] D. Jia, D. Zhang, and N. Li. Pulse waveform classification using support vector machine with gaussian time warp edit distance kernel. *Computational and Mathematical Methods in Medicine*, 2014, 2014. ISSN 17486718. doi: 10.1155/2014/947254.
- [128] T. Joachims. Text Categorization with Support Vector Machines: Learning with Many Relevant Features. *Proceedings of the 10th European Conference on Machine Learning ECML '98*, pages 137–142, 1998. ISSN 0945-1129. doi: 10.1007/BFb0026683.
- [129] R. R. Joon Yau Leong, Amir S. Patel. Using Electronic Health Records to Generate Phenotypes for Research. *Physiology & behavior*, 176(5):139–148, 2017. doi: 10.1002/cphg.80.Using.
- [130] H. J. Kam and H. Y. Kim. Learning representations for the early detection of sepsis with deep neural networks. *Computers in Biology and Medicine*, 89(August):248–255, 2017. ISSN 18790534. doi: 10.1016/j.compbiomed.2017.08.015. URL <https://doi.org/10.1016/j.compbiomed.2017.08.015>.
- [131] I. Kavakiotis, O. Tsave, A. Salifoglou, N. Maglaveras, I. Vlahavas, and I. Chouvarda. Machine Learning and Data Mining Methods in Diabetes Research. *Computational and Structural Biotechnology Journal*, 15:104–116, 2017. ISSN 20010370. doi: 10.1016/j.csbj.2016.12.005. URL <https://doi.org/10.1016/j.csbj.2016.12.005>.
- [132] E. Kennedy, W. Wiitala, R. Hayward, J. Sussman, X. Liu, and H. Leufkens. Personalised Medicine Strategy. *Medical care*, 51(3):e0174944, 2015. ISSN 1932-6203. doi: 10.1371/JOURNAL.PONE.0174944. URL <http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0174944>.

- [133] E. Keogh and C. A. Ratanamahatana. Exact indexing of dynamic time warping. *Knowledge and Information Systems*, 7(3):358–386, 2005. ISSN 02191377. doi: 10.1007/s10115-004-0154-9.
- [134] E. Keogh, K. Chakrabarti, M. Pazzani, and S. Mehrotra. Dimensionality Reduction for Fast Similarity Search in Large Time Series Databases. *Knowledge and Information Systems*, 3(3): 263–286, 2001. ISSN 0219-1377. doi: 10.1007/PL00011669. URL <http://link.springer.com/10.1007/PL00011669>.
- [135] H. Kharrazi, C. Wang, and D. Scharfstein. Prospective EHR-based clinical trials: The challenge of missing data. *Journal of General Internal Medicine*, 29(7):976–978, 2014. ISSN 15251497. doi: 10.1007/s11606-014-2883-0.
- [136] W. Kim, K. S. Kim, J. E. Lee, D. Y. Noh, S. W. Kim, Y. S. Jung, M. Y. Park, and R. W. Park. Development of novel breast cancer recurrence prediction model using support vector machine. *Journal of Breast Cancer*, 15(2):230–238, jun 2012. ISSN 17386756. doi: 10.4048/jbc.2012.15.2.230. URL <http://www.ncbi.nlm.nih.gov/pubmed/22807942><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC3395748><https://synapse.koreamed.org/DOIX.php?id=10.4048/jbc.2012.15.2.230>.
- [137] M. Kloft. Lp-Norm Multiple Kernel Learning. *Journal of Machine Learning Research*, 12(3): 953–997, 2011. ISSN 1532-4435. URL <http://eprints.pascal-network.org/archive/00009404/>.
- [138] W. C. Knowler, E. Barret-Connor, S. E. Fowler, R. F. Hamman, J. M. Lachin, E. A. Walker, and D. M. Nathan. Reduction in the Incidence of Type 2 Diabetes with Lifestyle Intervention or Metformin. *New England Journal of Medicine*, 346(6):393–403, 2015. ISSN 0028-4793. doi: 10.1056/NEJMoa012512. URL <http://www.nejm.org/doi/abs/10.1056/NEJMoa012512>.
- [139] F. Köpcke, B. Trinczek, R. W. Majeed, B. Schreiweis, J. Wenk, T. Leusch, T. Ganslandt, C. Ohmann, B. Bergh, R. Röhrig, M. Dugas, and H. U. Prokosch. Evaluation of data completeness in the electronic health record for the purpose of patient recruitment into clinical trials: A retrospective analysis of element presence. *BMC Medical Informatics and Decision Making*, 13(1):2–9, 2013. ISSN 14726947. doi: 10.1186/1472-6947-13-37.
- [140] K. Kourou, T. P. Exarchos, K. P. Exarchos, M. V. Karamouzis, and D. I. Fotiadis. Machine learning applications in cancer prognosis and prediction. *Computational and Structural Biotechnology Journal*, 13:8–17, 2015. ISSN 20010370. doi: 10.1016/j.csbj.2014.11.005. URL <http://linkinghub.elsevier.com/retrieve/pii/S2001037014000464>.
- [141] P. P. Kuksa. Biological sequence classification with multivariate string kernels. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 10(5):1201–1210, sep 2013. ISSN 15455963. doi: 10.1109/TCBB.2013.15. URL <http://ieeexplore.ieee.org/document/6475934/>.
- [142] G. Lanckriet and N. Cristianini. Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5:27–72, 2004. ISSN 15324435. doi: 10.1162/153244304322765649. URL <http://dl.acm.org/citation.cfm?id=1005334>.
- [143] T. A. Lasko, J. C. Denny, and M. A. Levy. Computational Phenotype Discovery Using Unsupervised Feature Learning over Noisy, Sparse, and Irregular Clinical Data. *PLoS ONE*, 8(6):e66341, 2013. ISSN 1932-6203. doi: 10.1371/journal.pone.0066341. URL <http://dx.plos.org/10.1371/journal.pone.0066341>.
- [144] F. Lau, C. Kuziemsy, M. Price, and J. Gardner. A review on systematic reviews of health information system studies. *Journal of the American Medical Informatics Association*, 17(6): 637–645, 2010. ISSN 10675027. doi: 10.1136/jamia.2010.004838.



- [145] N. Lee and A. F. Laine. Mining electronic medical records to explore the linkage between healthcare resource utilization and disease severity in diabetic patients. 2011. doi: 10.1109/HI SB.2011.34.
- [146] K. Lemström and E. Ukkonen. Including Interval Encoding into Edit Distance Based Music Comparison and Retrieval. 2000.
- [147] C. Leslie and W. S. Noble. Mismatch String Kernels for SVM Protein. *NIPS Proceedings*, 2003.
- [148] C. Leslie, E. Eskin, and W. S. Noble. The Spectrum Kernel: A String Kernel for SVM Protein Classification. *The Journal of Pharmacology and Experimental Therapeutics*, 142:257–264, 1963. ISSN 00223565.
- [149] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10(8):707–710, 1966. ISSN 00385689. doi: citeulike-article-id:311174.
- [150] A. LG and E. AT. Using Three Machine Learning Techniques for Predicting Breast Cancer Recurrence. *Journal of Health & Medical Informatics*, 04(02):2–4, 2013. ISSN 21577420. doi: 10.4172/2157-7420.1000124. URL [http://www.omicsonline.org/health-medical-informatics-abstract.php?abstract\\_{\\_}id=13087](http://www.omicsonline.org/health-medical-informatics-abstract.php?abstract_{_}id=13087).
- [151] H. Li and T. Jiang. A class of edit kernels for SVMs to predict translation initiation sites in eukaryotic mRNAs. *Journal of computational biology : a journal of computational molecular cell biology*, 12(6):702–18, 2005. ISSN 1066-5277. doi: 10.1089/cmb.2005.12.702. URL <http://www.ncbi.nlm.nih.gov/pubmed/16108712>{%}5Cn<http://online.liebertpub.com/doi/abs/10.1089/cmb.2005.12.702>.
- [152] H. Li, X. Li, M. Ramanathan, and A. Zhang. Identifying informative risk factors and predicting bone disease progression via deep belief networks. *Methods*, 69(3):257–265, 2014. ISSN 10959130. doi: 10.1016/j.ymeth.2014.06.011. URL <http://dx.doi.org/10.1016/j.ymeth.2014.06.011>.
- [153] L. Liu, W. Zuo, D. Zhang, N. Li, and H. Zhang. Combination of heterogeneous features for wrist pulse blood flow signal diagnosis via multiple kernel learning. *IEEE Transactions on Information Technology in Biomedicine*, 16(4):598–606, 2012. ISSN 10897771. doi: 10.1109/TITB.2012.2195188.
- [154] H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins. Text Classification using String Kernels. *Journal of Machine Learning Research*, 2:419–444, 2002. ISSN 0003-6951. doi: 10.1162/153244302760200687. URL <http://discovery.ucl.ac.uk/13443/>{%}5Cn[http://www.crossref.org/deleted\\_{\\_}DOI.html](http://www.crossref.org/deleted_{_}DOI.html).
- [155] G. Loosli, S. Canu, and C. S. Ong. Learning SVM in Krein Spaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(6):1204–1216, 2016. ISSN 01628828. doi: 10.1109/TPAMI.2015.2477830.
- [156] D. Lopresti and A. Tomkins. Block edit models for approximate string matching. *Theoretical Computer Science*, 181(1):159–179, 1997. ISSN 03043975. doi: 10.1016/S0304-3975(96)00268-X.
- [157] J. Louradour and K. Daoudi. Svm Speaker Verification Using a New Sequence Kernel. 3.
- [158] X. lu Xiong, R. xin Zhang, Y. Bi, W. hong Zhou, Y. Yu, and D. long Zhu. Machine Learning Models in Type 2 Diabetes Risk Prediction: Results from a Cross-sectional Retrospective Study in Chinese Adults. *Current Medical Science*, 39(4):582–588, 2019. ISSN 2523899X. doi: 10.1007/s11596-019-2077-4.

- [159] R. Luss and A. D’Aspremont. Support Vector Machine Classification with Indefinite Kernels. *Advances in Neural Information Processing Systems*, pages 953–960, 2008. ISSN 1867-2949. doi: 10.1007/s12532-009-0005-5. URL <http://papers.nips.cc/paper/3339-support-vector-machine-classification-with-indefinite-kernels.pdf>.
- [160] B. M. Marlin, D. C. Kale, R. G. Khemani, and R. C. Wetzel. Unsupervised pattern discovery in electronic health care data using probabilistic clustering models. *Proceedings of the 2nd ACM SIGHT symposium on International health informatics - IHI ’12*, page 389, 2012. doi: 10.1145/2110363.2110408. URL <http://dl.acm.org/citation.cfm?doid=2110363.2110408>.
- [161] P.-F. Marteau. Time Warp Edit Distance with Stiffness Adjustment for Time Series Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):306–318, 2009. doi: 10.1109/TPAMI.2008.76.
- [162] P.-F. Marteau and S. Gibet. On Recursive Edit Distance Kernels With Applications To Time Series Classification. *IEEE Transactions on Neural Networks and Learning Systems*, PP(6): 1–13, 2014. ISSN 2162237X. doi: 10.1109/TNNLS.2014.2333876.
- [163] F. J. Martin-Sanchez, V. Aguiar-Pulido, G. H. Lopez-Campos, N. Peek, and L. Sacchi. Secondary Use and Analysis of Big Data Collected for Patient Care. *IMIA Yearbook*, 26(1):28–37, 2017. ISSN 0943-4747. doi: 10.15265/IY-2017-008. URL <http://www.schattauer.de/index.php?id=1214&doi=10.15265/IY-2017-008>.
- [164] N. M. Maruthur, E. Tseng, S. Hutfless, L. M. Wilson, C. Suarez-Cuervo, Z. Berger, Y. Chu, E. Iyoha, J. B. Segal, and S. Bolen. Diabetes medications as monotherapy or metformin-based combination therapy for type 2 diabetes: A systematic review and meta-analysis. *Annals of Internal Medicine*, 164(11):740–751, 2016. ISSN 15393704. doi: 10.7326/M15-2650.
- [165] A. Marzal and E. Vidal. Computation of Normalized Edit Distance and Applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):926–932, 1993. ISSN 01628828. doi: 10.1109/34.232078.
- [166] A. McCallum and F. Pereira. A Conditional Random Field for Discriminatively- trained Finite-state String Edit Distance. *Conference on Uncertainty in AI (UAI)*, pages 388–395, 2005. URL <http://arxiv.org/abs/1207.1406> <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.127.8254> <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.127.8254&rank=1>.
- [167] X. H. Meng, Y. X. Huang, D. P. Rao, Q. Zhang, and Q. Liu. Comparison of three data mining models for predicting diabetes or prediabetes by risk factors. *Kaohsiung Journal of Medical Sciences*, 29(2):93–99, 2013. ISSN 1607551X. doi: 10.1016/j.kjms.2012.08.016.
- [168] L. Meyer, P. Bohme, I. Delbachian, P. Lehert, N. Cugnardey, P. Drouin, and B. Guerci. The Benefits of Metformin Therapy During Continuous Subcutaneous Insulin Infusion Treatment of Type 1 Diabetic Patients. *Diabetes Care*, 25(12):2153–2158, dec 2002. ISSN 0149-5992. doi: 10.2337/diacare.25.12.2153. URL <http://care.diabetesjournals.org/cgi/doi/10.2337/diacare.25.12.2153>.
- [169] R. Miotto, L. Li, B. A. Kidd, and J. T. Dudley. Deep Patient: An Unsupervised Representation to Predict the Future of Patients from the Electronic Health Records. *Scientific Reports*, 6 (May):1–10, 2016. ISSN 20452322. doi: 10.1038/srep26094.
- [170] E. Moradi, A. Pepe, C. Gaser, H. Huttunen, and J. Tohka. Machine learning framework for early MRI-based Alzheimer’s conversion prediction in MCI subjects. *NeuroImage*, 104:398–412, 2015. ISSN 10538119. doi: 10.1016/j.neuroimage.2014.10.002. URL <http://linkinghub.elsevier.com/retrieve/pii/S1053811914008131>.

- [171] M. Naffaa, V. Rosenberg, G. Chodick, V. Shalev, S. Tiosano, and H. Amital. SAT0074 Persistence with metformin treatment and onset of rheumatoid arthritis. In *Poster Presentations*, volume 76, pages 796.2–797. BMJ Publishing Group Ltd and European League Against Rheumatism, jun 2017. doi: 10.1136/annrheumdis-2017-eular.4226. URL <http://ard.bmj.com/lookup/doi/10.1136/annrheumdis-2017-eular.4226>.
- [172] H. Nasri and M. Rafeian-Kopaei. Metformin: Current knowledge, jul 2014. ISSN 17357136. URL <http://www.ncbi.nlm.nih.gov/pubmed/25364368><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4214027>.
- [173] National Institute for Health and Care Excellence. Type 2 diabetes in adults: management. NICE Guidelnes [NG28], 2015. URL <https://www.nice.org.uk/guidance/ng28/resources/type-2-diabetes-in-adults-management-1837338615493><https://www.nice.org.uk/guidance/ng28>.
- [174] National Institute for Health and Care Excellence. Context — Type 2 diabetes: prevention in people at high risk — Guidance — NICE, 2017. URL <https://www.nice.org.uk/guidance/ph38><https://www.nice.org.uk/guidance/ph38%0Ahttps://www.nice.org.uk/guidance/ph38/chapter/Context>.
- [175] G. Navarro. A guided tour to approximate string matching. *ACM Computing Surveys*, 33(1): 31–88, 2001. ISSN 03600300 (ISSN). doi: 10.1145/375360.375365. URL <http://www.scopus.com/inward/record.url?eid=2-s2.0-0345566149%0Ahttps://www.nice.org.uk/guidance/ph38/chapter/Context>
- [176] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48(3):443–453, 1970. ISSN 00222836. doi: 10.1016/0022-2836(70)90057-4.
- [177] M. Neuhaus and H. Bunke. Edit distance-based kernel functions for structural pattern classification. *Pattern Recognition*, 39(10):1852–1863, 2006. ISSN 00313203. doi: 10.1016/j.patcog.2006.04.012.
- [178] M. Neuhaus and H. Bunke. Automatic learning of cost functions for graph edit distance. *Information Sciences*, 177(1):239–247, 2007. ISSN 00200255. doi: 10.1016/j.ins.2006.02.013.
- [179] B. P. Nguyen, H. N. Pham, H. Tran, N. Nghiem, Q. H. Nguyen, T. T. Do, C. T. Tran, and C. R. Simpson. Predicting the onset of type 2 diabetes using wide and deep learning with electronic health records. *Computer Methods and Programs in Biomedicine*, 182:105055, 2019. ISSN 01692607. doi: 10.1016/j.cmpb.2019.105055. URL <https://doi.org/10.1016/j.cmpb.2019.105055>.
- [180] NHS. NHS England » General Practice Forward View (GPFV), 2017. URL <https://www.england.nhs.uk/gp/gpfv/%0Ahttps://www.england.nhs.uk/gp/gpfv/%0Ahttps://www.england.nhs.uk/publication/general-practice-forward-view-gpfv/>.
- [181] NHS England NHS Improvement. Securing Excellence in Primary Care ( GP ) Digital Services. The Primary Care (GP) Digital Services Operating Model 2019-2021. 2019. URL <https://www.england.nhs.uk/wp-content/uploads/2019/10/gp-it-operating-model-v4-sept-2019.pdf>.
- [182] NHSD. Read Codes - NHS Digital, 2019. URL <https://digital.nhs.uk/services/terminology-and-classifications/read-codes>.
- [183] B. Norgeot, B. S. Glicksberg, L. Trupin, D. Lituiev, M. Gianfrancesco, B. Oskotsky, G. Schmajuk, J. Yazdany, and A. J. Butte. Assessment of a Deep Learning Model Based on Electronic Health Record Data to Forecast Clinical Outcomes in Patients With Rheumatoid Arthritis. *JAMA network open*, 2(3):e190606, 2019. ISSN 25743805. doi: 10.1001/jamanetworkopen.2019.0606.

- [184] N. Nwegbu, S. Tirunagari, and D. Windridge. A novel kernel based approach to arbitrary length symbolic data with application to type 2 diabetes risk. *Scientific Reports*, 12(1):1–16, 2022. ISSN 2045-2322. doi: 10.1038/s41598-022-08757-1. URL <https://doi.org/10.1038/s41598-022-08757-1>.
- [185] C. Ochs, J. Geller, Y. Perl, Y. Chen, A. Agrawal, J. T. Case, and G. Hripcsak. A tribal abstraction network for SNOMED CT target hierarchies without attribute relationships. *Journal of the American Medical Informatics Association*, pages 628–639, 2015. doi: 10.1136/amiajnl-2014-003173.
- [186] T. Okuda, E. Tanaka, and T. Kasai. A Method for the Correction of Garbled Words Based on the Levenshtein Metric. *IEEE Transactions on Computers*, C-25(2):172–178, 1976. ISSN 00189340. doi: 10.1109/TC.1976.5009232.
- [187] J. Oncina and M. Sebhan. Learning stochastic edit distance: Application in handwritten character recognition. *Pattern Recognition*, 39(9):1575–1587, 2006. ISSN 00313203. doi: 10.1016/j.patcog.2006.03.011.
- [188] C. Ong and A. Smola. Machine learning using hyperkernels. *Proceedings of the International Conference on Machine Learning*, (Section 4):568–575, 2003. URL <http://www.aaai.org/Papers/ICML/2003/ICML03-075.pdf>.
- [189] C. S. Ong, X. Mary, S. Canu, and A. J. Smola. Learning with Non-Positive Kernels. *Association for Computing Machinery*, 2004. doi: 10.1145/1015330.1015443. URL <https://doi.org/10.1145/1015330.1015443>.
- [190] C. S. Ong, A. Smola, and B. Williamson. Learning the Kernel with Hyperkernels. *Journal of Machine Learning Research*, 6:1043–1071, 2005. ISSN 1532-4435. URL <http://eprints.pascal-network.org/archive/00002012/>.
- [191] C. S. Ong, S. Canu, and G. Loosli. Technical report : SVM in Krein spaces. 2013. URL <https://hal.science/hal-00869658>.
- [192] B. Oommen. Constrained string editing. *Information Sciences*, 40(3):267–284, 1986. ISSN 0020-0255. doi: [https://doi.org/10.1016/0020-0255\(86\)90061-7](https://doi.org/10.1016/0020-0255(86)90061-7). URL <https://www.sciencedirect.com/science/article/pii/0020025586900617>.
- [193] B. J. Oommen and K. Zhang. The Normalized String Editing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(6):669–672, 1996. doi: 10.1109/34.506420.
- [194] M. Panahiazar, V. Taslimitehrani, N. Pereira, and J. Pathak. Using EHRs and Machine Learning for Heart Failure Survival Analysis. *Studies in Health Technology and Informatics*, 216:40–44, 2015. ISSN 18798365. doi: 10.3233/978-1-61499-564-7-40.
- [195] W. Paper. Toward a National Framework for the Secondary Use of Health. *Journal of the American Medical Informatics Association : JAMIA*, 14(1):1–9, 2007. doi: 10.1197/jamia.M2273.Introduction.
- [196] J. Paul, R. D’Ambrosio, and P. Dupont. Kernel methods for heterogeneous feature selection. *Neurocomputing*, 169:187–195, 2015. ISSN 18728286. doi: 10.1016/j.neucom.2014.12.098. URL <http://dx.doi.org/10.1016/j.neucom.2014.12.098>.
- [197] P. L. Peissig, V. Santos Costa, M. D. Caldwell, C. Rottscheit, R. L. Berg, E. A. Mendonca, and D. Page. Relational machine learning for electronic health record-driven phenotyping. *Journal of Biomedical Informatics*, 52:260–270, 2014. ISSN 15320464. doi: 10.1016/j.jbi.2014.07.007. URL <http://dx.doi.org/10.1016/j.jbi.2014.07.007>.

- [198] M. Perez-Nieves, S. Kabul, U. Desai, J. I. Ivanova, N. Y. Kirson, A. K. Cummings, H. G. Birnbaum, R. Duan, D. Cao, and I. Hadjiyianni. Basal insulin persistence, associated factors, and outcomes after treatment initiation among people with type 2 diabetes mellitus in the US. *Current medical research and opinion*, 32(4):669–80, 2016. ISSN 1473-4877. doi: 10.1185/03007995.2015.1135789. URL <http://www.tandfonline.com/doi/full/10.1185/03007995.2015.1135789>. URL <http://www.ncbi.nlm.nih.gov/pubmed/26703951>.
- [199] S. Perveen, M. Shahbaz, K. Keshavjee, and A. Guergachi. Prognostic Modeling and Prevention of Diabetes Using Machine Learning Technique. *Scientific Reports*, 9(1):1–9, 2019. ISSN 20452322. doi: 10.1038/s41598-019-49563-6. URL <http://dx.doi.org/10.1038/s41598-019-49563-6>.
- [200] R. Pivovarov, D. J. Albers, J. L. Sepulveda, and N. Elhadad. Identifying and mitigating biases in EHR laboratory tests. *Journal of Biomedical Informatics*, 51:24–34, 2014. ISSN 15320464. doi: 10.1016/j.jbi.2014.03.016. URL <http://dx.doi.org/10.1016/j.jbi.2014.03.016>.
- [201] S. Pscherer, E. Chou, F.-w. Dippel, W. Rathmann, K. Kostev, and S.-a. D. Gmbh. Treatment persistence after initiating basal insulin in type 2 diabetes patients : A primary care. *Primary Care Diabetes*, 9(5):377–384, 2015. ISSN 1751-9918. doi: 10.1016/j.pcd.2015.01.011. URL <http://dx.doi.org/10.1016/j.pcd.2015.01.011>.
- [202] A. Rakotomamonjy, F. R. Bach, S. Canu, and Y. Grandvalet. SimpleMKL. *Journal of Machine Learning Research*, 9:2491–2521, 2008. ISSN 15324435.
- [203] C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning*. MIT Press, 2006. ISBN 026218253X. URL <http://www.gaussianprocess.org/gpml/>.
- [204] L. Rasmy, W. J. Zheng, H. Xu, D. Zhi, Y. Wu, N. Wang, H. Wu, X. Geng, and F. Wang. A study of generalizability of recurrent neural network-based predictive models for heart failure onset risk using a large and heterogeneous EHR data set. *Journal of Biomedical Informatics*, 84(May):11–16, 2018. ISSN 15320464. doi: 10.1016/j.jbi.2018.06.011. URL <https://doi.org/10.1016/j.jbi.2018.06.011>.
- [205] C. Ratanamahatana and E. Keogh. Everything you know about dynamic time warping is wrong. *Third Workshop on Mining Temporal and Sequential Data*, pages 22–25, 2004. ISSN 00903493. doi: 10.1097/01.CCM.0000279204.24648.44. URL [http://spoken-number-recognition.googlecode.com/svn/trunk/docs/Dynamictimewarping/DTW\\_{\\_}myths.pdf](http://spoken-number-recognition.googlecode.com/svn/trunk/docs/Dynamictimewarping/DTW_{_}myths.pdf).
- [206] C. A. Ratanamahatana and E. Keogh. Making Time-series Classification More Accurate Using Learned Constraints. *Proceedings of the 2004 SIAM International Conference on Data Mining*, pages 11–22, 2004. doi: 10.1137/1.9781611972740.2. URL <http://epubs.siam.org/doi/abs/10.1137/1.9781611972740.2>.
- [207] K. Riesen and H. Bunke. Approximate graph edit distance computation by means of bipartite graph matching. *Image and Vision Computing*, 27(7):950–959, 2009. ISSN 02628856. doi: 10.1016/j.imavis.2008.04.004. URL <http://dx.doi.org/10.1016/j.imavis.2008.04.004>.
- [208] E. S. Ristad and P. N. Yianilos. Learning string edit distance. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(5):522–532, 1998. ISSN 01628828. doi: 10.1109/34.682181. URL <http://arxiv.org/abs/cmp-lg/9610005>.
- [209] F. S. Roque, P. B. Jensen, H. Schmock, M. Dalgaard, M. Andreatta, T. Hansen, K. Søbey, S. Bredkjær, A. Juul, T. Werge, L. J. Jensen, and S. Brunak. Using electronic patient records to discover disease correlations and stratify patient cohorts. *PLoS Computational Biology*, 7(8), 2011. ISSN 1553734X. doi: 10.1371/journal.pcbi.1002141.

- [210] A. Saenz, I. Fernandez-Esteban, A. Mataix, M. Ausejo Segura, M. Roqué i Figuls, and D. Moher. Metformin monotherapy for type 2 diabetes mellitus. In A. Saenz, editor, *Cochrane Database of Systematic Reviews*. John Wiley & Sons, Ltd, Chichester, UK, jul 2005. doi: 10.1002/14651858.CD002966.pub3. URL <http://doi.wiley.com/10.1002/14651858.CD002966.pub3>.
- [211] H. Saigo, J. P. Vert, N. Ueda, and T. Akutsu. Protein homology detection using string alignment kernels. *Bioinformatics*, 20(11):1682–1689, 2004. ISSN 13674803. doi: 10.1093/bioinformatics/bth141.
- [212] A. Sanfeliu, A. Sanfeliu, and K. S. Fu. A Distance Measure Between Attributed Relational Graphs for Pattern Recognition. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-13(3):353–362, 1983. ISSN 21682909. doi: 10.1109/TSMC.1983.6313167.
- [213] M. Sattlecker, N. Stone, and C. Bessant. Current trends in machine-learning methods applied to spectroscopic cancer diagnosis. *TrAC - Trends in Analytical Chemistry*, 59:17–25, 2014. ISSN 18793142. doi: 10.1016/j.trac.2014.02.016. URL <http://dx.doi.org/10.1016/j.trac.2014.02.016>.
- [214] J. H. Scarpello. Review: Optimal dosing strategies for maximising the clinical response to metformin in type 2 diabetes. *The British Journal of Diabetes & Vascular Disease*, 1(1):28–36, 2001. ISSN 1474-6514. doi: 10.1177/14746514010010010501.
- [215] G. Seni, V. Kripasundar, and R. K. Srihari. Generalizing edit distance to incorporate domain information: Handwritten text recognition as a case study. *Pattern Recognition*, 29(3):405–414, 1996. ISSN 00313203. doi: 10.1016/0031-3203(95)00102-6.
- [216] D. Shapira and J. A. Storer. Edit distance with move operations. *Journal of Discrete Algorithms*, 5(2 SPEC. ISS.):380–392, 2007. ISSN 15708667. doi: 10.1016/j.jda.2005.01.010.
- [217] B. Shickel, P. J. Tighe, A. Bihorac, and P. Rashidi. Deep EHR: A Survey of Recent Advances in Deep Learning Techniques for Electronic Health Record (EHR) Analysis. *Biomed Health Inform.*, 176(3):139–148, 2017. doi: 10.1016/j.physbeh.2017.03.040.
- [218] B. Shickel, P. J. Tighe, A. Bihorac, and P. Rashidi. Deep EHR: A Survey of Recent Advances in Deep Learning Techniques for Electronic Health Record (EHR) Analysis. *IEEE Journal of Biomedical and Health Informatics*, 22(5):1589–1604, sep 2018. ISSN 21682194. doi: 10.1109/JBHI.2017.2767063. URL <http://www.ncbi.nlm.nih.gov/pubmed/29989977http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC6043423https://ieeexplore.ieee.org/document/8086133/>.
- [219] A. Shimoda, D. Ichikawa, and H. Oyama. Using machine-learning approaches to predict non-participation in a nationwide general health check-up scheme. *Computer Methods and Programs in Biomedicine*, 163:39–46, 2018. ISSN 18727565. doi: 10.1016/j.cmpb.2018.05.032. URL <https://doi.org/10.1016/j.cmpb.2018.05.032>.
- [220] H. Shimodaira, K.-i. Noma, M. Nakai, and S. Sagayama. Dynamic time-alignment kernel in support vector machine. In T. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems*, volume 14. MIT Press, 2001. URL <https://proceedings.neurips.cc/paper/2001/file/a869ccbcb9568808b8497e28275c7c8-Paper.pdf>.
- [221] K. Shin. Alignment kernels based on a generalization of alignments. *IEICE Transactions on Information and Systems*, E97-D(1):1–10, 2014. ISSN 17451361. doi: 10.1587/transinf.E97.D.1.
- [222] G. Simonyi and T. Ferenci. A metformin-monoterápia és a szitagliptin/metformin fix kombináció egyéves perzisztenciája. *Orvosi Hetilap*, 157(16):618–622, apr 2016. ISSN 0030-6002. doi: 10.1556/650.2016.30423. URL <http://www.ncbi.nlm.nih.gov/pubmed/27063429http://www.akademiai.com/doi/abs/10.1556/650.2016.30423>.

- [223] A. Singh, G. Nadkarni, O. Gottesman, S. B. Ellis, E. P. Bottinger, and J. V. Guttag. Incorporating temporal EHR data in predictive models for risk stratification of renal function deterioration. *Journal of Biomedical Informatics*, 53:220–228, 2015. ISSN 15320464. doi: 10.1016/j.jbi.2014.11.005. URL <http://dx.doi.org/10.1016/j.jbi.2014.11.005>.
- [224] J. Smoller. The Use of Electronic Health Records for Psychiatric Phenotyping and Genomics. *American Journal of Medical Genetics Part B Neuropsychiatric Genetics*, 177(1):139–148, 2017. doi: 10.1002/ajmg.b.32548.
- [225] SNOMED International. SNOMED - 5-Step Briefing, 2021. URL <https://www.snomed.org/snomed-ct/five-step-briefing>.
- [226] C. Soguero-Ruiz, K. Hindberg, I. Mora-Jiménez, J. L. Rojo-Álvarez, S. O. Skrøvseth, F. Godtliebsen, K. Mortensen, A. Revhaug, R. O. Lindsetmo, K. M. Augestad, and R. Jenssen. Predicting colorectal surgical complications using heterogeneous clinical data and kernel methods. *Journal of Biomedical Informatics*, 61:87–96, 2016. ISSN 15320464. doi: 10.1016/j.jbi.2016.03.008. URL <http://dx.doi.org/10.1016/j.jbi.2016.03.008>.
- [227] Y. J. Son, H. G. Kim, E. H. Kim, S. Choi, and S. K. Lee. Application of support vector machine for prediction of medication adherence in heart failure patients. *Healthcare Informatics Research*, 16(4):253–259, 2010. ISSN 20933681. doi: 10.4258/hir.2010.16.4.253.
- [228] M. Song and A. Rudniy. Detecting Duplicate Biological Entities Using Markov Random Field-Based Edit Distance. *2008 IEEE International Conference on Bioinformatics and Biomedicine*, (3):457–460, 2008. doi: 10.1109/BIBM.2008.34. URL <http://ieeexplore.ieee.org/document/4684939/>.
- [229] S. Sonnenburg, G. Rátsch, and C. Schäfer. A general and efficient multiple kernel learning algorithm. *Nips*, (1), 2005.
- [230] B. C. Stuart, X. Shen, C. C. Quinn, N. Brandt, P. Roberto, F. E. Loh, F. Hendrick, C. Kim, X. Huang, and S. Rajpathak. Proximal Predictors of Long-Term Discontinuation with Noninsulin Antihyperglycemic Agents. *Journal of managed care & specialty pharmacy*, 22(9):1019–1027, 2016. ISSN 2376-1032. doi: 10.18553/jmcp.2016.22.9.1019. URL <http://www.ncbi.nlm.nih.gov/pubmed/27574743>.
- [231] J. Sun and C. Reddy. Big data analytics for healthcare. In *In Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, page 1525, 2013. ISBN 9781450321747. doi: 10.1145/2487575.2506178. URL <http://dl.acm.org/citation.cfm?id=2487575.2506178>.
- [232] J. Sun, J. Hu, D. Luo, M. Markatou, F. Wang, S. Edabollahi, S. E. Steinhubl, Z. Daar, and W. F. Stewart. Combining knowledge and data driven insights for identifying risk factors using electronic health records. *AMIA ... Annual Symposium proceedings / AMIA Symposium. AMIA Symposium*, 2012:901–910, 2012. ISSN 1942597X.
- [233] J. Sun, C. D. Mcnaughton, P. Zhang, A. Perer, A. Gkoulalas-Divanis, J. C. Denny, J. Kirby, T. Lasko, A. Saip, and B. A. Malin. Predicting changes in hypertension control using electronic health records from a chronic disease management program. *Journal of the American Medical Informatics Association*, 21(2):337–344, 2014. ISSN 10675027. doi: 10.1136/amiajnl-2013-002033.
- [234] J. Tarhio. *On Using q-Gram Locations in Approximate String Matching*, volume 979. 1995. ISBN 978-3-540-60313-9. doi: 10.1007/3-540-60313-1. URL <http://link.springer.com/10.1007/3-540-60313-1>.

- [235] J. Tian, B. Yu, D. Yu, and S. Ma. Missing data analyses: A hybrid multiple imputation algorithm using Gray System Theory and entropy based on clustering. *Applied Intelligence*, 40(2):376–388, 2014. ISSN 0924669X. doi: 10.1007/s10489-013-0469-x.
- [236] S. Tian, S. Mu, and C. Yin. Sequence-similarity kernels for SVMs to detect anomalies in system calls. *Neurocomputing*, 70(4-6):859–866, 2007. ISSN 09252312. doi: 10.1016/j.neucom.2006.10.017.
- [237] S. Tirunagari, S. Bull, and N. Poh. Automatic Classification of Irregularly Sampled Time Series With Unequal Lengths: a Case Study on Estimated Glomerular Filtration Rate.
- [238] E. E. Tripoliti, T. G. Papadopoulos, G. S. Karanasiou, K. K. Naka, and D. I. Fotiadis. Heart Failure: Diagnosis, Severity Estimation and Prediction of Adverse Events Through Machine Learning Techniques. *Computational and Structural Biotechnology Journal*, 15:26–47, 2017. ISSN 20010370. doi: 10.1016/j.csbj.2016.11.001.
- [239] S. Uchida and H. Sakoe. A survey of elastic matching techniques for handwritten character recognition. *IEICE Transactions on Information and Systems*, E88-D(8):1781–1790, 2005. ISSN 17451361. doi: 10.1093/ietisy/e88-d.8.1781.
- [240] M. van Zaanen. ABL: Alignment-Based Learning. *COLING*, pages 961–967, 01 2001. doi: 10.3115/992730.992785. URL <http://arxiv.org/abs/cs/0104006>.
- [241] S. Vella, L. Buetow, P. Royle, S. Livingstone, H. M. Colhoun, and J. R. Petrie. The use of metformin in type 1 diabetes: a systematic review of efficacy. *Diabetologia*, 53(5):809–820, may 2010. ISSN 0012-186X. doi: 10.1007/s00125-009-1636-9. URL <http://link.springer.com/10.1007/s00125-009-1636-9>.
- [242] R. A. Wagner and M. J. Fischer. The String-to-String Correction Problem. *Journal of the ACM*, 21(1):168–173, 1974. ISSN 00045411. doi: 10.1145/321796.321811. URL <http://portal.acm.org/citation.cfm?doid=321796.321811>.
- [243] F. Wang. Adaptive semi-supervised recursive tree partitioning: The ART towards large scale patient indexing in personalized healthcare. *Journal of Biomedical Informatics*, 55:41–54, 2015. ISSN 15320464. doi: 10.1016/j.jbi.2015.01.009. URL <http://linkinghub.elsevier.com/retrieve/pii/S1532046415000118>.
- [244] T. Wang, H. Su, and J. Li. Neurocomputing DWS-MKL : Depth-width-scaling multiple kernel learning for data classification. *Neurocomputing*, 411:455–467, 2020. ISSN 0925-2312. doi: 10.1016/j.neucom.2020.06.039. URL <https://doi.org/10.1016/j.neucom.2020.06.039>.
- [245] Y. Wang, R. Chen, J. Ghosh, J. C. Denny, A. Kho, B. A. Malin, and J. Sun. Rubik : Knowledge Guided Tensor Factorization and Completion for Health Data Analytics Categories and Subject Descriptors.
- [246] C. Watkins. Dynamic Alignment Kernels. *Advances in Large Margin Classifiers*, (January): 39–50, 1999. ISSN 10495258. doi: 10.1.1.40.4778. URL <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.40.4778{&}rep=rep1{&}type=pdf>.
- [247] N. R. Waugh, D. Shyangdan, S. Taylor-Phillips, G. Suri, and B. Hall. Screening for type 2 diabetes: A short report for the National Screening Committee. *Health Technology Assessment*, 17(35):1–89, 2013. ISSN 13665278. doi: 10.3310/hta17350.
- [248] J. Wei. Markov Edit Distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(3):311–321, 2004. ISSN 01628828. doi: 10.1109/TPAMI.2004.1262315.



- [249] L. Wei, E. Keogh, A. Mafra-Neto, and R. J. Abbott. Efficient Query Filtering for Streaming Time Series with Applications to Semi Supervised Learning of Time Series Classifiers. *Working Paper*, 2005.
- [250] N. G. Weiskopf, G. Hripcsak, S. Swaminathan, and C. Weng. Defining and measuring completeness of electronic health records for secondary use. *Journal of Biomedical Informatics*, 46(5):830–836, 2013. ISSN 15320464. doi: 10.1016/j.jbi.2013.06.010. URL <http://dx.doi.org/10.1016/j.jbi.2013.06.010>.
- [251] N. G. Weiskopf, C. Weng, and N. G. Weiskopf. Methods and dimensions of electronic health record data quality assessment : enabling reuse for clinical research. pages 144–151, 2013. doi: 10.1136/amiajnl-2011-000681.
- [252] D. Windridge and M. Bober. *A Kernel-Based Framework for Medical Big-Data Analytics*, volume 8401, pages 197–208. 01 2014. ISBN 978-3-662-43967-8. doi: 10.1007/978-3-662-43968-5\_11.
- [253] D. Windridge, V. Mottl, A. Tatarchuk, and A. Elisseyev. The neutral point method for kernel-based combination of disjoint training data in multi-modal pattern recognition. *Multiple Classifier Systems*, pages 13–21, 2007. ISSN 03029743. URL <http://www.springerlink.com/index/X5861513MU330464.pdf>.
- [254] V. Wottschel, D. Alexander, P. Kwok, D. Chard, M. Stromillo, N. De Stefano, A. Thompson, D. Miller, and O. Ciccarelli. Predicting outcome in clinically isolated syndrome using machine learning. *NeuroImage: Clinical*, 7:281–287, 2015. ISSN 22131582. doi: 10.1016/j.nicl.2014.11.021. URL <http://linkinghub.elsevier.com/retrieve/pii/S2213158214001880>.
- [255] J. J. Wright, R. M. Hahn, A. Wivel, A. A. Martin, K. Prussia, and U. Kingdom. Self-reported Barriers to Adherence and Persistence to Treatment With Injectable Medications for Type 2 Diabetes. *Clinical Therapeutics*, 38(7):1653–1664.e1, 2016. ISSN 0149-2918. doi: 10.1016/j.clinthera.2016.05.009. URL <http://dx.doi.org/10.1016/j.clinthera.2016.05.009>.
- [256] G. Wu, E. Y. Chang, and Z. Zhang. An analysis of transformation on non-positive semidefinite similarity matrix for kernel machines. *Icml*, 8, 2005. doi: 10.1016/j.jaut.2015.04.002. Widely. URL <http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=FF310C7FBC79D95FA507E9FB93D48994?doi=10.1.1.133.4077&rep=rep1&type=pdf><http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.133.4077&rep=rep1&type=pdf>[5Cnpapers2://publication/uuid/0C628F77-94](http://publication/uuid/0C628F77-94).
- [257] J. Wu, J. Roy, and W. F. Stewart. Prediction Modeling Using EHR Data. *Medical Care*, 48(6):S106–S113, 2010. ISSN 0025-7079. doi: 10.1097/MLR.0b013e3181de9e17.
- [258] P.-y. Wu, C.-w. Cheng, C. D. Kaddi, J. Venugopalan, R. Hoffman, M. D. Wang, and S. Member. – Omic and Electronic Health Record Big Data Analytics for Precision Medicine. 64(2):263–273, 2017.
- [259] X. Xi, E. Keogh, C. Shelton, L. Wei, and C. A. Ratanamahatana. Fast time series classification using numerosity reduction. *Proceedings of the 23rd international conference on Machine learning - ICML '06*, pages 1033–1040, 2006. doi: 10.1145/1143844.1143974. URL <http://portal.acm.org/citation.cfm?doid=1143844.1143974>.
- [260] C. Xiao, E. Choi, and J. Sun. Opportunities and challenges in developing deep learning models using electronic health records data: A systematic review. *Journal of the American Medical Informatics Association*, 25(10):1419–1428, 2018. ISSN 1527974X. doi: 10.1093/jamia/ocy068.

- [261] X. Xu, I. W. Tsang, and D. Xu. Soft margin multiple kernel learning. *IEEE Transactions on Neural Networks and Learning Systems*, 24(5):749–761, 2013. ISSN 2162237X. doi: 10.1109/TNNLS.2012.2237183.
- [262] Z. Xu, R. Jin, H. Yang, I. King, and M. R. Lyu. Simple and efficient multiple kernel learning by group lasso. *International Conference on Machine Learning (ICML)*, pages 1191–1198, 2010.
- [263] H. Xue and S. Chen. Neurocomputing Discriminality-driven regularization framework for indefinite kernel machine. *Neurocomputing*, 133:209–221, 2014. ISSN 0925-2312. doi: 10.1016/j.neucom.2013.11.016. URL <http://dx.doi.org/10.1016/j.neucom.2013.11.016>.
- [264] C. Ye, T. Fu, S. Hao, Y. Zhang, O. Wang, B. Jin, M. Xia, M. Liu, X. Zhou, Q. Wu, Y. Guo, and C. Zhu. Prediction of Incident Hypertension Within the Next Year : Prospective Study Using Statewide Electronic Health Records and Machine Learning. 20:1–18, 2018. doi: 10.2196/jmir.9268.
- [265] D. Zhang, W. Zuo, D. Zhang, Y. Li, and N. Li. Gaussian ERP kernel classifier for pulse waveforms classification. *Proceedings - International Conference on Pattern Recognition*, pages 2736–2739, 2010. ISSN 10514651. doi: 10.1109/ICPR.2010.670.
- [266] J. Zhao, P. Papapetrou, L. Asker, and H. Boström. Learning from heterogeneous temporal data in electronic health records. *Journal of Biomedical Informatics*, 65:105–119, 2017. ISSN 15320464. doi: 10.1016/j.jbi.2016.11.006. URL <http://dx.doi.org/10.1016/j.jbi.2016.11.006>.
- [267] B. Zheng, S. W. Yoon, and S. S. Lam. Breast cancer diagnosis based on feature extraction using a hybrid of K-means and support vector machine algorithms. *Expert Systems with Applications*, 41(4 PART 1):1476–1482, 2014. ISSN 09574174. doi: 10.1016/j.eswa.2013.08.044. URL <http://dx.doi.org/10.1016/j.eswa.2013.08.044>.
- [268] T. Zheng, W. Xie, L. Xu, X. He, Y. Zhang, M. You, G. Yang, and Y. Chen. A machine learning-based framework to identify type 2 diabetes through electronic health records. *International Journal of Medical Informatics*, 97:120–127, 2017. ISSN 18728243. doi: 10.1016/j.ijmedinf.2016.09.014. URL <http://dx.doi.org/10.1016/j.ijmedinf.2016.09.014>.