Machine Learning in the Model Space for Metabolomics Time Series

in an adrenal steroid hormone study

by

XINYUE CHEN

Lead Supervisor: Prof. Peter Tino
Co-Supervisor: Dr. Yuan Shen and Prof. David Smith

A thesis submitted to the University of Birmingham for the degree of
DOCTOR OF PHILOSOPHY

School of Computer Science
University of Birmingham
February 2023

Abstract

Learning in the model space (LiMS) aims to represent each complex data subject such as sparse and/or noisy time series with an appropriate model, or a full posterior distribution over models. LiMS approaches include mechanistic information on how the data is generated in the machine learning model-building stage. Hence, it can improve the interpretability of chosen machine learning tools. This thesis proposes a new topographic mapping approach as well as a time series classification application in the model space. Both of them are demonstrated on a real-world data set of measurements taken on subjects in an adrenal steroid hormone study.

Topographic visualisation methods such as self-organisation maps are important tools in data mining. In order to cluster and visualise sparse and/or noisy time series data, a novel self-organising map directly formulated in the model space termed as SOMiMS is proposed, together with an extension of generative topographic mapping (GTM) to the model space. Both maps are demonstrated on the adrenal steroid hormone data set with a good degree of separation of conditions. Compared to classic approaches in the signal space, they take the mechanistic information into account by providing interpretable readily data visualisations and parameter plots in the form of heat maps.

In biomedical settings, time series classification is one of the most important techniques to improve the accuracy of disease diagnosis. The time series classification in the model space is developed not only to improve the diagnosis accuracy but also to provide mechanistic

and biomedical model interpretability. It is applied to the adrenal steroid hormone data set showing satisfying classification performance in both signal and model space. Two classifier models, support vector machine (SVM) and logistic regression are employed. In addition, a hybrid model which significantly improves the accuracy is also created. Through feature selection, important time periods (signal space) and model parameters (model space) are extracted, which are crucial and valuable information from the biomedical point of view. In the data preprocessing stage, the missing value and initial value problems, which are two common problems of biomedical data are solved by using the univariate Gaussian process and adjoint method. Analyses and evaluations are concluded along with mechanistic and biomedical knowledge and case studies of some additional subjects.

Acknowledgement

First and foremost, I would like to express my deepest gratitude to my lead supervisor Prof. Peter Tino for his unlimited support and unconditional guidance during my Ph.D. journey. I would also like to extend my sincere thanks to my second supervisor Dr. Yuan Shen for his assistance and insightful suggestions at every stage of my research. I thank my co-supervisor Prof. David Smith. Without their encouragement and constant feedback this research would not be achievable.

In addition, I would like to acknowledge our medical collaborators Eder Zavala, Georgina Russell, Krasimira Tsaneva-Atanasova, Stafford Lightman, and Thomas Upton, who provided precious real data set and biomedical knowledge for my research. Thanks should also go to Prof. Sanjiv Narayan and the School of Computer Science, University of Birmingham for their financial support in undertaking my Ph.D. study.

In the last place, special thanks to my family and friends for their encouragement and support all through my Ph.D. journey.

Contents

List of Figures

List of Tables

List of abbreviation

Aldo	Aldosterone
CCS	Corticosterone
EM	Expectation Maximization
GP	Gaussian process
GTM	Generative topographic mapping
KNN	K-nearest neighbor
LiMS	Learning in the model space
MLE	Maximum likelihood estimation
PrimAldo	Primary Aldosterone
SOMiMS	Self-organisation map
SVM	Support vector machine

Chapter 1 Introduction

The research topic of this doctoral research is machine learning in the model space for metabolomics time series data applied to a real data set of adrenal steroid hormones. Each steroid hormone subject is a multivariate time series that is noisy and sparsely sampled. In order to handle such complex data, machine learning in the model space (LiMS) framework is proposed. In LiMS, each time series is represented by a model. Then, the learning is formulated in the space of models. In this adrenal steroid hormone study, a parametric inferential mechanistic model is provided, which is flexible enough to represent the variety of data items. Compared to traditional machine learning approaches working in the signal space, LiMS approaches are not only capable of dealing with sparse and/or noisy time series data with comparable performance, but are also able to provide biomedical insights and interpretations. Combining with the mechanistic model, time series clustering and classification methodologies are developed.

This chapter first gives a brief research background introduction. Then, research motivations and objectives are given, following with the outline of the thesis at the end of this chapter.

1.1 Research background and motivations

Time series is a very common type of data, containing a sequence of data whose measurements are taken over time (usually at regular time intervals), which are called observations. When the observed space is multidimensional, the time series becomes multivariate, otherwise it is univariate. Time series data are widely available in different fields, for example finance, science, healthcare, etc. due to the development of the internet, technology, and digital healthcare. Thus, the importance of time series analysis is expected to grow significantly in the coming years [1].

1.1.1 Time series clustering and classification

Usually, time series analysis tasks involve classification, clustering, regression, and forecasting. However, the time series analysis is facing unique challenges. In real-life settings, time series measurements obtained are noisy and/or sparse. A time series is said to be sparsely sampled when the intervals between successive observations are long. Such time series data arise when sampling processes are difficult and complex especially in domains including medicine, biology, astronomy [2][3][4]. Also, some of them may contain substantial observational gaps [5]. Hence, it is not feasible to use existing machine learning algorithms for example support vector machine, random forest, logistic regression, etc. on such raw time series directly. This thesis focuses on time series clustering and classification.

Clustering is a unsupervised data mining technique which places similar data into ho-

mogeneous groups without knowing the groups' definitions. Time series clustering is a special type of clustering [6]. Essentially, a time series is treated as dynamic data because value(s) of each point of a time series is/are one or more observations that are made chronologically. Although each time series contains a large number of data points, it can also been treated as a single object. It is advantageous to cluster such complex data because it leads to discovery of interesting patterns in time series datasets [7]. According to [6], there are four main reasons to develop time series clustering. Firstly, time series datasets involve valuable information, which can be obtained through pattern discovery and clustering is the most common way to uncover these patterns on time series datasets. Secondly, most time series datasets are too large and complex to be handled well by human inspectors. Therefore, users prefer to deal with structured datasets rather than original larger datasets. As a result, clustering is helpful to generate such structured datasets as a set of groups of similar time series by aggregation of data in non-overlapping clusters. Thirdly, time series clustering is useful for exploratory data analysis or as a pre-processing step for other data mining tasks. Finally, the visualization of time series data based on clustering is able to help users to understand the structure of data quickly.

Consider a dataset with n time series data $D = \{X_1, X_2, ..., X_n\}$. Time series clustering refers to the unsupervised process to partition D into $C = \{C_1, C_2, ..., C_k\}$, in such a way that homogeneous time series are grouped together based on a certain similarity measure. C_i indicates a cluster, where $D = \cup_{i=1}^{k} C_i$ and $C_i \cap C_j = \varnothing$ for $i \neq j$.

However, time series clustering is challenging because firstly, time series data are nor-

mally high dimensional, which makes handling these data difficult for many clustering algorithms and also slows down the process of clustering [8][9]. Secondly, the similarity measure that are used to make the clusters is a major challenge, especially for whole sequence matching where whole lengths of time series are considered during distance calculation because time series data are naturally noisy and include outliers and shifts, at the other hand the length of time series varies [6][10].

Time series classification involves building predictive models that output a target label from inputs of sequential observations across some time period [5][11]. In time series classification, an instance is a pair $\{\boldsymbol{x}, y\}$ with j observations $(x_1, x_2, ..., x_j)$ and y is the discrete class variable with i possible values. A time series dataset contains n such instances $D = [\{\boldsymbol{x}_1, y_1\}, \{\boldsymbol{x}_2, y_2\}, ..., \{\boldsymbol{x}_n, y_n\}]$. Hundreds of time series classification algorithms have been proposed in recent years [12] and they can be divided into two camps: deep-learning models and non-deep-learning models [11]. Deep-learning approaches such as Convolutional Neural Networks (CNNs) and Deep Neural Networks (DNNs) have shown many successful applications recently [12], while they also have challenges in training, hyperparameter tuning, model complexity decisions, etc. Compared to deep learning approaches, non-deep-learning methods are generally easier to train, optimise and deploy [11]. This thesis is going to focus on non-deep-learning classification approaches.

Typically, the time series classification has three main steps, which are signal transformation or preprocessing, modelling, and classification (Fig. 1.1). The first step normally involves missing value imputation, normalization, adjustments, etc. In the second step,

different types of algorithms are developed and they can be summarised into feature engineering and selection, statistical modeling, distance-based, index development, and shape-based methods. In the third step, the classifier model is first selected, then the hyperparameter tuning, model training, and validation can be done.

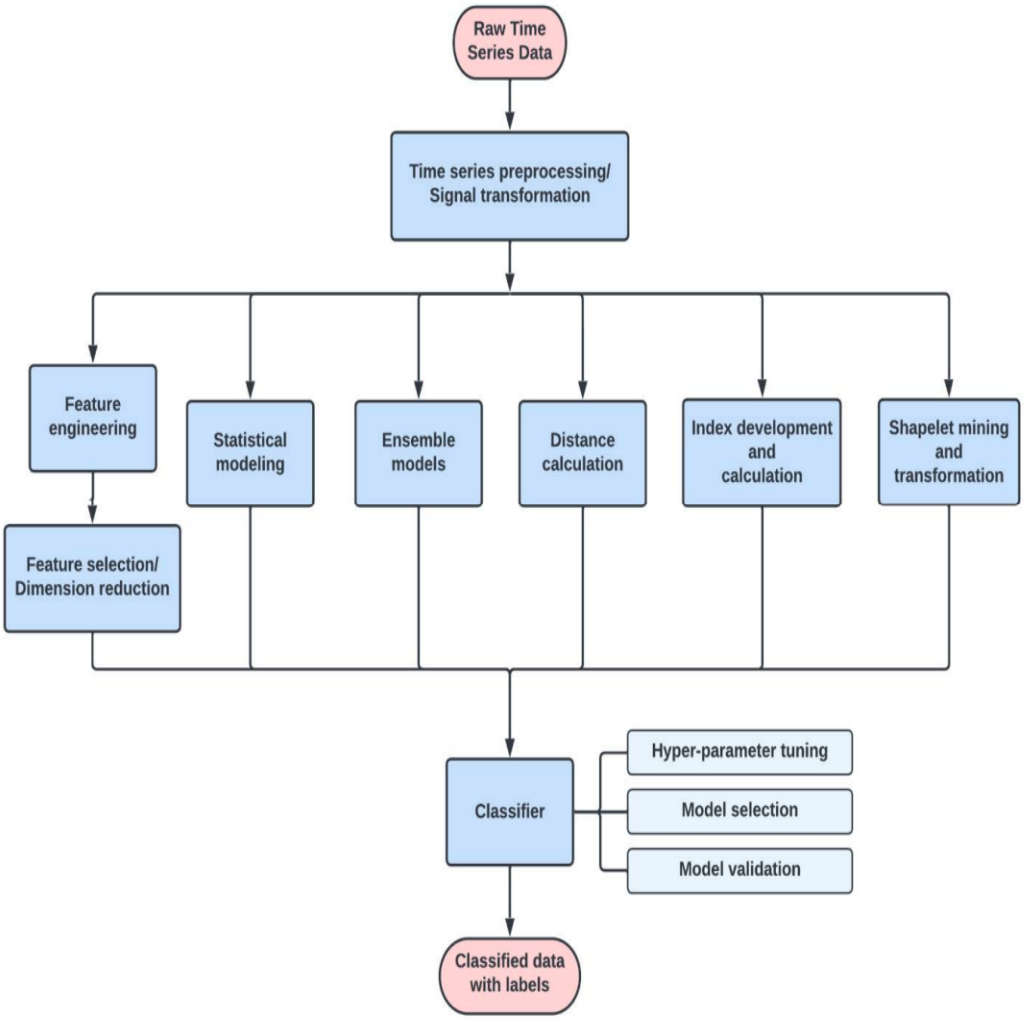


Figure 1.1: Time series classification steps

In the first step, filtering is the most common preprocessing method for noise reduction and artifact removal. Methods such as smoothing, segmentation, and re-sampling are also used depending on different research tasks. Besides, other common methods are using Fourier transform for signal decomposition[13], feature extraction, using wavelet trans-

form to decompose the original signal into different frequency bands [14], etc.

In the second step of time series classification modelling, the feature engineering approach is most widely used. A vector of common feature engineering techniques is summarised in Table 1.1. After using certain feature engineering techniques, new features extracted from the time series data are used in the following step as representatives of the original data.

Ensemble approaches are the combination of multiple models. Typically, the final prediction is made by the majority vote of ensembles. These methods may or may not require an additional feature engineering step. Distance-based models simply calculate the differences in time series data based on different similarity measures. Shape-based models are characterised by mining or comparing shapes or patterns in a time series or sequence vector. State-space models refer to the construction of a probability model. In addition, other models for example using statistical modelling and designing a composite index or metric based on data-driven metrics or domain knowledge, are also proposed.

Table 1.1: Different types of feature engineering techniques

Feature type	Description
Amplitude	Amplitude features refer to how distant a signal's values from 0
Frequency	Frequency features describe the properties of the Fourier transform
Stationarity	Stationarity represents the consistency of signal properties over time, such as mean and variance
Entropy	Entropy measures the number of states of a system or the ability to probabilistically determine the next state of system
Variability	Variability measures how similar in value each of the measurements in a signal are
Linearity	Linearity features quantity how much a system changes with a constant rate
Correlation	Correlation features describe how dependent a signal is on previous state
Plot-based	Features related to properties of a certain graph

1.1.2 Time series in biomedical applications

With the development of the technology (wearable devices like smart watches) and the digitisation of healthcare systems, there has been an emerging increase of biomedical time series data sets and researches using those data, including accelerometry for activity recognition, polysomnography (PSG) for sleep tracking [15], electroencephalogram

(EEG) for brainwave tracking [16], electrocardiogram (ECG) for cardiovascular dysfunction screening [17], etc. As a result, there is a desperate need to develop some data mining or classification approaches in order to extract or detect useful information from those biomedical time series data. This will lead to the development of more reliable and accurate methods for diagnosing, monitoring and screening and provide significant benefits to the healthcare system, which not only could save money and time, but also more importantly could save lives.

Compared to time series data in other fields, biomedical time series data collected from human subjects have some unique obstacles, which leverage time series modeling techniques. One of the biggest challenges is the small data set size. Usually, biomedical data sets contain just a small number of human subjects because of the effort and resources required for the data collection. This makes it even more difficult for using deep learning models because most of them are data-hungry [11]. Thus, non-deep learning models are more suitable for small-size biomedical time series data sets. Another challenge is the individual differences among human subjects, meaning that models perform well on some individuals may have negative outcomes for others.

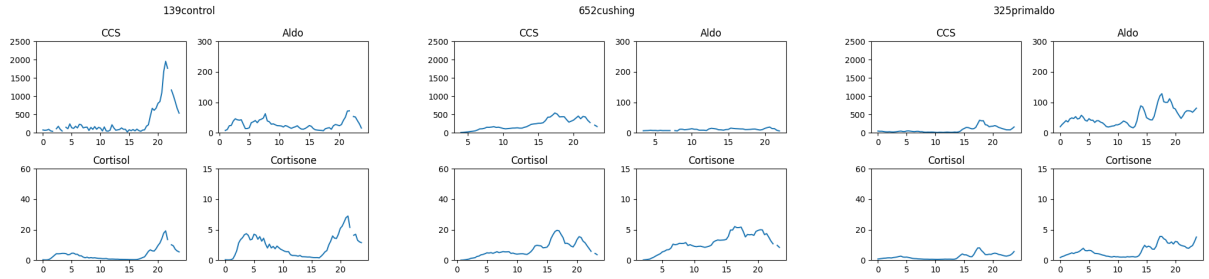
Moreover, model interpretability is an important aspect that has to be taken into account, especially for biomedical applications. Accurate and validated interpretation of models could explain potential insights from the biomedical point of view [11]. Some models with a methodology of interpretation built in and information based on domain knowledge are able to provide reasonable model interpretability. However, it is still a challenge for most

models, even if they have great performance.

1.1.3 Learning in the model space for noisy or/and sparse time series

Under the learning in the model space framework, each data item (e.g. time series) is represented by a model that "explains "it. Then, learning is formulated in the space of models. In order to involve the certain domain knowledge and provide appropriate model interpretability, the model here refers to inferential mechanistic model, which is corresponding to the input time series dataset. Usually, such inferential mechanistic model with free parameters is flexible to represent variety of data items and is also sufficiently constrained to avoid overfitting.

This research demonstrates the learning in the model space approach on the adrenal steroid hormone dataset. This dataset contains three possible clusters, which are Control, Cushing's and Primary Aldosteronism (PrimAldo). Each time series in the dataset is multivariate and has four metabolites in the data feature space – CCS, Aldo, Cortisol and Cortisone. Examples of time series trajectories in three conditions are given in Figure 1.2. The length of each time series may vary because of missing values at random time points. The inferential biomedical model corresponding to this dataset is provided by medical experts with thirteen free parameters and in the form of coupled ordinary differential equations (ODEs).



(a) Control (b) Cushing's (c) PrimAldo

Figure 1.2: Trajectories of Control 1.2a, Cushing's1.2b and PrimAldo 1.2c

Leaning in the model space is not only able to take the model interpretability into account, but is capable of dealing with challenges of time series analysis mentioned in section 1.1.1. In this thesis, the adrenal steroid hormones time series are sparse and noisy. Compared to other time series data which have observations every second, for example financial data, adrenal steroid hormones time series have observations every twenty minutes, which is a very long interval compared to every second. Also, most medical data are with high noise and small sample sizes because of the complex sampling procedure (e.g. collecting blood samples or body fluid) [2]. In this research, learning in the model space approaches estimate the noise level from the data using Gaussian Process. Moreover, each adrenal steroid hormones time series contains random missing values, which leads to the length of each time series is different. To deal with challenges discussed above, learning in the model space approaches transfer each multivariate time series into model parameters, which is a thirteen parameter vector. It is easier and faster to handle model parameters because they are normally lower dimensional than time series signals. Also, the calculation of the similarity among model parameter vectors is more straightforward because they have the same length.

1.1.4 Research motivations

An accurate diagnosis is critical to prevent wasting precious time on the wrong course of treatment. Machine learning especially the classification has the potential to help improve the diagnosis results, while it is still in the early stages of implementation. Moreover, most existing machine learning approaches diagnose diseases only based on patients' symptoms. They are unable to explain the symptoms by determining the causes of certain diseases [18]. To solve this problem, the model interpretability has to be taken into account when a machine learning model is built. However, most common machine learning models are not capable of providing appropriate model interpretability [19]. Thus, this research develops a LiMS approach, which not only can provide significant performance but also can take mechanistic and biomedical knowledge into account.

Real-life biomedical data are usually sparse and/or noisy time series data. It is impractical to simply apply machine learning algorithms such as logistic regression or support vector machine to such raw time series data. Hence, this research introduces a feature generation approach in the model space. Each time series is transferred to a suitable model, represented by a vector of model parameters.

Besides, data visualisation, which is a special approach of clustering is useful and important. It is helpful to discover some underlying facts, trends and patterns quickly [6]. Self organising maps and generative topographic mapping are two common methods used for visualisation and exploratory data analysis of high dimensional datasets. However, classic self organising map and generative topographic mapping can only be applied to

data with same length [20] [21]. Thus, this research proposes both self organising map and generative topographic mapping in the model space to handle noisy or/and sparse time series with different length. Also, both of them offer parameter plots, which can provide underlying biomedical insights for medical people.

1.2 Research objectives and contributions

1.2.1 Research objectives

The research objectives are to:

- Accomplish the data preprocessing for the adrenal steroid hormone data set in particular dealing with substantial missingness in the data and transform them into the model space by representing each subject using the inferential mechanistic model (Chapter 5)
- Develop a self-organisation mapping algorithm in the model space for model based clustering and visualisation of sparse and/or noisy time series data (Chapter 4)
- Extend the generative topographic mapping into the model space and compare it with the self-organisation mapping algorithm in the model space (Chapter 4)
- Apply the time series classification in the model space approach to the adrenal steroid hormone data set and design an appropriate experiment methodology for it. (Chapter 5)

1.2.2 Research contributions

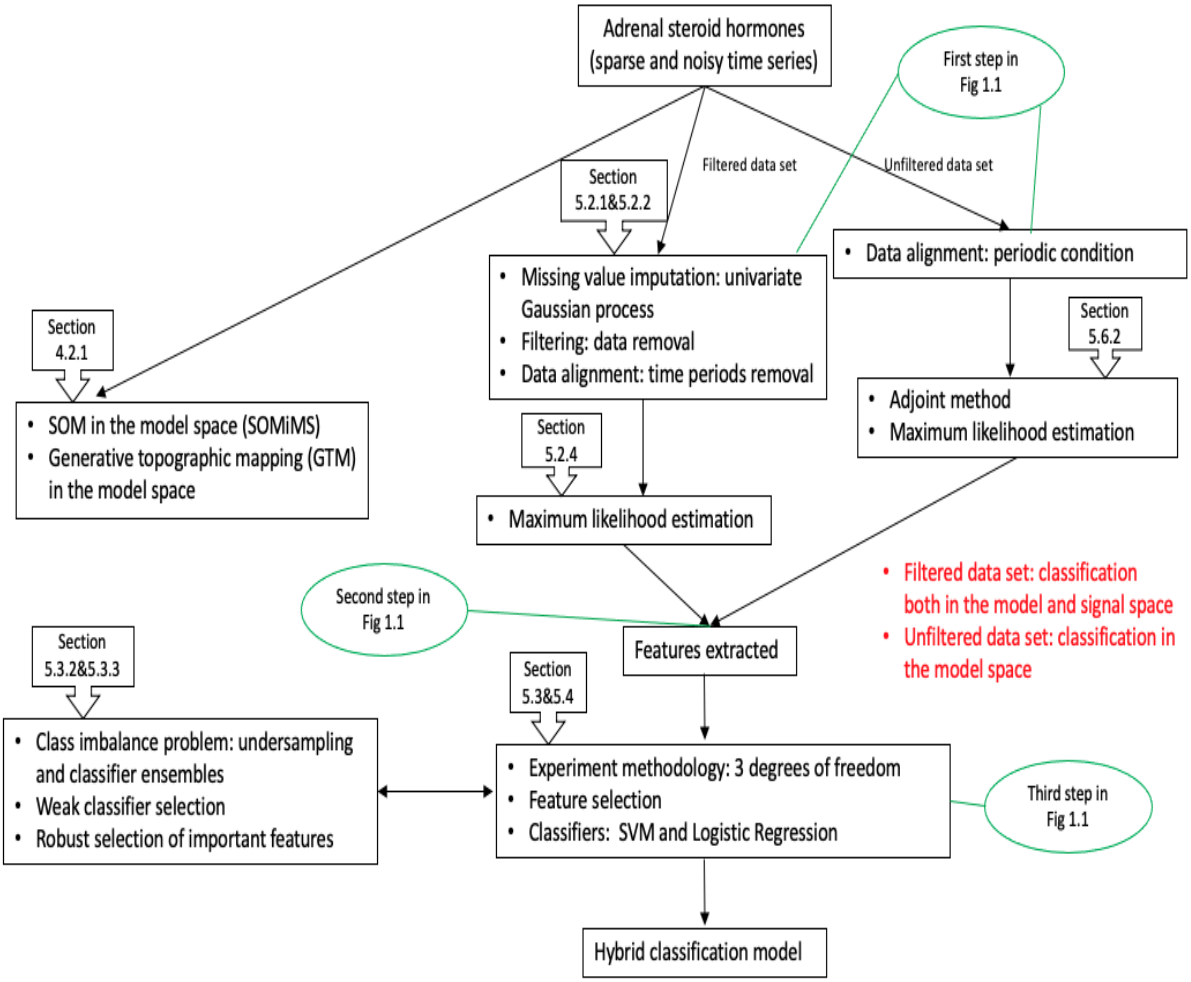


Figure 1.3: Research contributions *Filtered data set contains 140 data subjects in good condition with only few missing values. Unfiltered data set involves 270 data subjects and some of them have substantial observation gaps.*

A flowchart of research contributions has been provided in Fig.1.3 and details are also described as:

- The adrenal steroid hormone data set has been cleaned and each data subject has been transferred to a model parameter vector (representative of a certain inferential mechanistic model) using maximum likelihood estimation (MLE) framework with

constraints.

- Self-organisation map (SOM) has been formulated directly in the model space named SOM in the model space (SOMiMS). People can visualize complex data such as sparse and/or noisy time series in a two-dimensional map using SOMiMS. It has been demonstrated on the adrenal steroid hormone data set.

- Generative topographic mapping (GTM) has been extended to the model space. It can also visualise a set of sparse time series. Mapping results of SOMiMS and extended GTM have been compared on the same data set.

- Parameter plots of SOMiMS and extended GTM, which can help readily interpret the topographic mappings from the mechanistic and biomedical points of view have been provided.

- The univariate Gaussian process has been used to impute missing values in the filtered adrenal steroid hormone data set.

- The gradient based optimisation algorithm is adopted to obtain the corresponding model parameter vector for each time series data by maximising the likelihood. The adjoint method has been applied to efficiently calculate the total derivative of likelihood, which is required by the The gradient based optimiser.

- The adrenal steroid hormone dataset is class-imbalance with more health control subjects than two other conditions. The repeated undersampling with replacement and classifier ensembles have been used to solve the class imbalance problem.

- An importance-based feature selection algorithm has been developed based on clas-

sifier ensembles with accuracy greater than 95%. It is a robust way to do feature selection.

- An experiment methodology for the time series classification model has been designed. It is based on the 3 degrees of design freedom (signal versus model space; full versus partial mechanistic model; full versus subset feature space).

- Based on the experimental design, a hybrid classification model has been developed. It combines classification models in the signal and model space for the filtered data set (both classification models in the model space for the unfiltered data set) and classification performance has been improved. Also, it has been verified by a set of new data, which are not used in previous training and testing.

- The interpretation of classification and feature selection results combined with the biomedical information has been presented.

- Conference paper: Chen, Xinyue, et al. "SOMiMS-Topographic Mapping in the Model Space."International Conference on Intelligent Data Engineering and Automated Learning. Springer, Cham, 2021.

- Other papers: written up together with medical collaborators.

1.3 Outline

Chapter 2 is a literature review of the research. Both time series classification and clustering approaches are reviewed. Time series clustering algorithms are discussed and compared, especially the self-organising map and its extensions such as generative to-

pographic mapping. Machine learning applications in multivariate steroid data are also reviewed.

Chapter 3 demonstrates the clinical and biomedical background of this research. Adrenal glands, adrenal steroid synthesis pathways, and their related hormones are introduced. Three types of disorders, together with their causes and symptoms are discussed.

Chapter 4 proposes a self-organizing mapping formulated directly in the model space and extends the generative topographic mapping to the model space to deal with noisy and/or sparsely sampled time series. Results are demonstrated on a real data set of adrenal steroid hormones. Parameter plots where values learnt for each mechanistic model parameter across the grid are shown as heat maps. K-nearest neighbor (KNN) classification is applied to map projections to quantify the degree of the separation on mappings. Interpretations of topographic mappings and parameter plots are given from the mechanistic point of view.

Chapter 5 applies time series classification in the model space approach to the adrenal steroid hormone data set. The univariate Gaussian process used to impute missing values is explained. The class imbalance problem in the classification is solved through repeated undersampling and classifier ensembles. Importance plots are created for the feature selection. A novel experiment methodology is proposed based on 3 degrees of design freedom. A hybrid model is developed by combining models in both signal and model space. Classification and feature selection results are demonstrated on both filtered and unfiltered

data sets, together with interpretations bonding with clinical and biomedical knowledge.

Chapter 6 is a conclusion of the thesis. The outcomes and key findings of the research are listed, together with its limitations. At last, the possible future work is discussed.

Chapter 2 Literature review

2.1 Introduction

This chapter presents a literature review of the relevant subjects of the research. Firstly, different approaches of time series classification are reviewed, compared, and discussed. Secondly, time series clustering is described together with existing clustering algorithms for time series. Two topographic visualization tools, self-organisation maps, and generative topographic mapping are presented. Finally, a brief review of machine learning applications on multivariate steroid data is provided.

2.2 Time series classification

Major approaches for time series classification can be divided into three categories, feature-based, distanced-based, and model-based approaches [22].

2.2.1 Feature-based and distance-based approaches

Feature-based approaches aim to transfer a time series into vector features and apply conventional classification methods. These features can be local patterns for example short sequence segments [23] or global ones based on time-frequency and wavelet decomposition method [24].

Distance-based approaches do classification through measuring the distance between time series pairs. Once a distance measurement is applied, some existing classification approaches, for example, the K-nearest neighbor classifier (KNN) can be used. The choice of distance function is significant to the classification performance. Euclidean distance is widely used for simple time series classification [22]. However, Euclidean distance is not applicable for time series with variable lengths. Then dynamic time warping distance (DTW) is developed to deal with this problem. In DTW two time series are aligned based on some criteria so that the distance of two time series can be calculated [25].

2.2.2 Model-based approaches

Feature-based and distance-based approaches mentioned above are not capable of dealing with time series that are noisy and/or sampled sparsely because they are not able to take observational noise into account, which makes the models prone to overfitting very easily [26] and because there can be substantial uncertainty about the underlying temporal processes due to the sparsity of observations [2]. Also, they are not able to integrate experts' knowledge about the underlying processes with classifiers. Alternatively, model-based methods are adopted to classify time series, for example, Hidden Markov Model (HMM), which is used in biological sequences [27] and speech recognition [28]. Whereas those model-based methods assume time series in one class are generated by a single underlying model M. Once a class of time series is given, M models the probability distribution of time series in this class. Normally, the probability distributions are described by some parameters. Those parameters are learnt through training. Then a new time series will be assigned to the class with the highest likelihood or the highest posterior probability if

class priors are available.

However, one single model might not be sufficient to represent each time series in the according class because there are infinitely many models which can fit the time series well when the time series is very noisy and sparse[26]. To overcome this problem, it is necessary to represent each time series by individual models [29]. In this case, the classifier works on individual models, which correspond to individual time series. This approach is called LiMS.

Most LiMS time series classification approaches represent each time series as point estimates of model parameters. Point estimates could be treated as feature vectors and any vector-based classification can be used here. For instance, in [30], long time series data with variable lengths are transferred into a high dimensional dynamical feature space through reservoir computation models. Then each time series is represented by the corresponding read-out mappings of the generic fixed dynamical reservoir. In this task, for all time series, the underlying dynamic reservoir will be the same, while the read-out models can capture the differences in each time series. Then, the read-out parameters are treated as feature vectors, which are used for time series classification. Also, Brodersen et al. use a dynamic causal model (DCM) to replace high dimensional functional magnetic resonance imaging (fMRI) data. Each fMRI data is represented by a low dimensional vector of parameter estimates, which are used to classify DCMs. Both approaches above are LiMS methods using point estimates to do classification for time series.

In addition, other LiMS methods employ model distances directly in the parameter space, which is the geodesic on the model manifold. In these approaches, the parameter space is treated as a non-linear metric and the metric is learned on the underlying manifold [29]. The intrinsic nature of the underlying processes or the constraints imposed on the models could generate a non-linear structure. To get geodesic distances, the underlying metric tensor field in the parameter space can be reconstructed first. A general framework based on pullback metric is presented to learn discriminative metric tensors in the space of Linear Dynamical Systems (LDS) and HMM respectively. The manifold structure in the parameter space is induced by stability constraints on the LDS parameters or by normalisation constraints on the HMM parameters [31][32].

Moreover, another type of LiMS approach is generated in the framework of kernel machines. The adopted kernels can illustrate the underlying non-linear structure in the model space via useful distance functions, although they do not recover the whole underlying metric tensor field [29]. Generally, employed kernels have been developed to operate on the probability distributions. In [33], kernels derived from the symmetric Kullback-Leibler (KL) divergence between two distributions are proposed to classify multimedia objects. Also, a probabilistic kernel based on the KL divergence is developed to do the classification of visual processes, which are modeled with spatiotemporal autoregressive models [34]. In addition, the probability product kernel (PPK) is proposed [35]. In PPK, a general inner product is calculated as the integral of the dot product of pairs of probability distributions. The PPK can evaluate all exponential family models like Gaussians and multinomials. The PPK is also computable for latent distributions such as HMM, linear

dynamical systems, and mixture models [35]. Bhattacharyya kernels are a special version of PPK, which is related to Hellinger's distance between distributions. However, generally, KL and PPK kernels' computation is very expensive and they are only computable for simple classes of dynamical systems because they can include infinite-dimensional integral over all possible state trajectories [29]. In contrast to PPK kernels, Binet-Cauchy kernels based on the Binet-Cauchy theorem are defined as a dot product in the trajectory space [36]. Taking the initial conditions of the dynamical systems into account is an advantage of Binet-Cauchy kernels.

Finally, there are two well-known kernels, the fisher kernel, and the autoregressive (AR) kernel, although they are outside the LiMS framework because no individual model is corresponding to the individual data object. The fisher kernel uses one fixed model and then each time series is represented by a tangent vector in the tangent space of that model [37]. AR kernel is developed based on the vector autoregressive (VAR) model. Each time series is described by the feature, which is the likelihood of parameters in the VAR model given that time series object. To compare two time series, the product of their likelihood features, which is weighted by a prior distribution over the VAR parameters is employed [38].

2.3 Time series clustering

Clustering belongs to unsupervised learning and is a technique to place similar data into homogeneous clusters without knowing the label information of the data [39]. Time series clustering is a special case of clustering as time series are complex and usually real-world

time series data are high dimensional and multivariate [6], which causes a speed decrease in the clustering process. Besides, the similarity measures are the key challenge for the time series clustering because most real-world time series data are noisy and with different lengths [6][10].

2.3.1 Time series clustering algorithms

The choice of certain clustering algorithms is dependent on the research objectives, which can be pattern discovery, prediction, visualisation, etc. Time series clustering algorithms are classified into five broad groups, which are demonstrated in Fig.2.1.

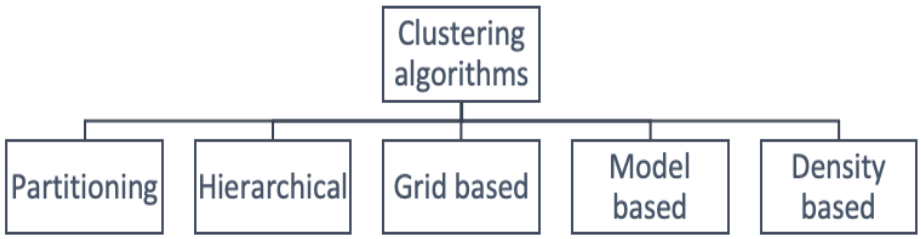


Figure 2.1: Groups of clustering algorithms

The partitioning clustering approach makes a set of subjects assigned into k groups and each group has at least one subject. K-means [40] is one of the most typical partitioning clustering algorithms by minimising the total distance from a certain cluster center between all subjects in this cluster. Usually, the cluster center in k-Means is defined as a mean vector of subjects in the cluster. Another algorithm in the partitioning group is the k-Medoids (PAM) algorithm [41], which uses one of the nearest objects to the cluster

center as the prototype. Both k-Means and k-Medoids require the number of clusters k to be pre-assigned. However, it is not practical for many applications and also for time series data [42]. Time series are very large or high dimensional so it is hard to check and tune the cluster number k. Nonetheless, k-Means and k-Medoids are very fast compared to other clustering algorithms for example hierarchical clustering. It makes them still widely used in many time series applications [10][43][44].

Partitioning clustering approaches can be built in 'hard'or 'fuzzy 'manner. Fuzzy c-Means and fuzzy c-Medoids are algorithms based on soft clusters, where each subject has a degree of membership in each cluster [45][46]. There are various time series clustering applications using fuzzy partitioning methods. A fuzzy variant is used to cluster subject motions in a collection of videos by adopting an EM-based optimization algorithm and a HMMs mixture of time series [47]. Both [48] and [49] used fuzzy c-Means for MRI data and speaker verification respectively.

To sum up, the definition of cluster centers together with their updating method is crucial thing for partitioning approaches. Thus, they are more suitable for time series with the same length whose similarity to each other is more straightforward to be measured.

Hierarchy clustering method [41] creates a hierarchy of clusters using agglomerative or divisive algorithms. Compared to partitioning clustering, hierarchy clustering has several advantages. Firstly, it does not need the number cluster pre-defined. In addition, it can be used to cluster time series with different lengths by employing Dynamic Time Warping

(DTW) as the similarity measure [50]. However, due to its quadratic computational complexity, hierarchy clustering is not suitable for large time series data set [19]. Furthermore, the quality of hierarchy clustering is not competitive as the clusters cannot be adjusted after splitting. Thus, hierarchy clustering is normally combined with other algorithms to address this issue [6].

In grid-based clustering approaches, the data space is quantised into a finite number of grid cells and the clustering is developed in this grid. The main advantage of these approaches is their fast process time because they are dependent on the number of cells in each dimension rather than the number of data subjects. However, according to [6], related work of grid-based time series clustering is rare in the literature.

In density-based clustering, clusters refer to a contiguous region with high point density, separated from subjects with low density (considered as noise or outliers). Density-based spatial clustering of applications with noise (DBSCAN) [51] is the most famous density-based clustering approach. DBSCAN has two parameters, *eps* (the neighbourhood) and *minPts* (a minimum number). For each cluster, the neighbourhood has to have a minimum number of points. Based on *eps* and *minPt*, the core and border points are assigned to a certain cluster. However, the main issue of DBSCAN is that it cannot treat clusters with various densities. Accordingly, Ordering Points To Identify the Clustering Structure (OPTICS) is developed to address this problem [52]. It is an advantage of density-based clustering to be able to distinguish noisy data. However, it is not widely used for time series because of its high complexity.

In most model-based clustering approaches, each cluster is presented by a model and data are fitted to that model. However, one single model may not be enough to well represent each time series (especially sparse and/or noise time series). Then a LiMS approach can be employed here by representing each time series with an individual model. Typically, people use SOM [20] to deal with the time series clustering problem [6]. SOM is a special type of neural network that is used for model-based clustering. It is also a useful tool for data analysis and visualization for high-dimensional data [53]. Classic SOM is not suitable for time series data with unequal length because SOM needs to define the dimension of the weight vector. Additionally, few researches use model-based clustering of time series that are composed of Gaussian mixed models [54], polynomial models [55], HMM [56] etc. Generally, there are two drawbacks of model-based clustering. Firstly, model parameters are required and the selection of models is subjective which may lead to bad data fitting. As a result, the clustering result would be biased or even inaccurate. Secondly, it is time-consuming on a large data set.

Some algorithms discussed above are constrained by the unequal length of time series. Generally, this is a similarity measure problem, which can be addressed by transferring time series into feature or model vectors before doing clustering. For example, LiMS can represent each time series by a set of model parameters, even if time series are very sparse and/or noisy. In addition, most clustering algorithms are not able to visualize data especially when input data are high dimensional, except SOM. Thus, the combination of LiMS and SOM would be a good and novel concept for the clustering and visualisation

of sparse and/or noisy time series.

2.3.2 Self-organisation map and its extensions

SOM [20] is a valuable technique for the visualisation and analysis of multivariate data. It has become an inspiration of numerous extensions. SOM is a type of neural network and able to project a high dimensional data space to a projection space with a lower dimension. The projection space is a grid of nodes/neurons related to a weight vector and normally the space is two-dimensional for visualisation purposes. To form the final map, all nodes in the grid compete and cooperate with each other. The first step is calculating the responses of all nodes to input data and the certain node with the greatest response is the winner node. In classic SOM, the response is measured using the Euclidean distance. Then, the winner node is updated towards the input data based on a learning rate in order to have better responses in future iterations. In addition, the neighbourhood (decided by a neighbourhood function) of the winner node is also updated according to the distance from the winner. Both neighbourhood and learning will decay after each iteration. Although SOM is a powerful algorithm and has achieved various successful applications in practice, it also has certain limitations, for instance, it has no proper cost function or any general proof of convergence. There is no theoretical support for choosing neighbourhood function parameters and learning rate parameters.

Accordingly, the generative topographic map (GTM) [21] has been proposed as a probabilistic reformulation of SOM. The GTM believes that in reality the high dimensional data only live on a 'noisy 'lower-dimensional manifold. Thus, it is appropriate to model a

given data set by optimising model parameters to match the model-generated data in the low-dimensional manifold in the distribution sense. GTM uses a Gaussian mixture model constrained by the lower dimensional manifold and model parameters can be optimised by Expectation-Maximization (EM) algorithm. GTM is a form of a nonlinear latent variable model. It maps the latent space to the data space and then the mapping is inverted to a posterior distribution in the latent space for visualisation purposes.

In addition, there is an increasing interest to extend SOM/GTM into the model space. Compared to classic SOM/GTM, which are mainly designed to operate in a vectorial data space, formulating SOM/GTM in the model space will be helpful to handle data with complex structures, e.g. [57][58][59]. In [57] the authors introduce the SOMAR (Self-Organising mixture autoregressive) model. In SOMAR, AR models that are utilised to model foreign exchange (FX) rates are employed here as the components in the construction of the topological mixture model. By using a constrained mixture of discrete HMM, [58] extends GTM to the model space. For visualization of tree-structured data, the work in [58] is extended again by [59] to the space of Hidden Tree Models. Besides clustering, SOM/GTM and their extensions are widely used for data visualisation. Particularly, their extensions in the model space are capable to handle data with complex structures.

2.4 Machine learning applications on steroid data

Artificial intelligence especially machine learning applications are becoming more widely recognised in biology and medicine. In [60], the author has demonstrated that machine

learning algorithms (random forest (RF), weighted-subspace random forest (WSRF), and extreme gradient boosted tree (XGBT)) in a data-driven manner can provide high performance for predicting abnormal urine steroid profiles with implications for clinical applications. The authors of [61] show the identification and classification of primary aldosteronism combined with machine learning techniques using steroid profiles. Four different machine learning algorithms are used to identify specific steroid combinations that can provide optimal segregation of patients with and without primary aldosteronism. Also, the random forest algorithms are trained to provide automatic detection of recurrent adrenocortical carcinoma on 19 steroid markers measured by GC-MS in a given 24-hour urine [62]. Usually, steroid data are multivariate and multidimensional. Despite the advantages of multidimensional analyses such as providing more data information, they still face challenges including redundancy in information across different feature dimensions [63]. Accordingly, feature selection is a way to select significant and relevant features from multidimensional data. It can be applied before or together with machine learning algorithms to improve machine learning performance. However, there have been few steroid profiling studies that have appropriately implemented machine learning strategies due to limitations of sample size or clinical study design.

Chapter 3 Clinical and biomedical background

3.1 Introduction

This chapter provides an overall introduction of the clinical and biomedical background related to the adrenal steroid hormone study, which is helpful to understand the whole project. An explanation of the adrenal glands, together with their related adrenal steroid synthesis pathways and some important hormones is provided. In addition, this chapter explains adrenal gland disorders and presents two common types of disorders, together with their corresponding hormones, symptoms, causes, etc.

3.2 Adrenal steroid synthesis pathways and hormones

Clinically, the adrenal glands (Fig.3.1) are two small triangular-shaped glands located on the top of both kidneys. They produce different types of hormones to help regulate people's immune system, blood pressure, metabolism, and other essential functions. Thus, adrenal glands play an important role in steroid and catecholamine synthesis. Adrenal glands have two distinct parts, an outer adrenal cortex, and an inner adrenal medulla [64], which are responsible for producing different hormones. This chapter focuses on the adrenal cortex, which is related to the mechanistic model and real data set used later.

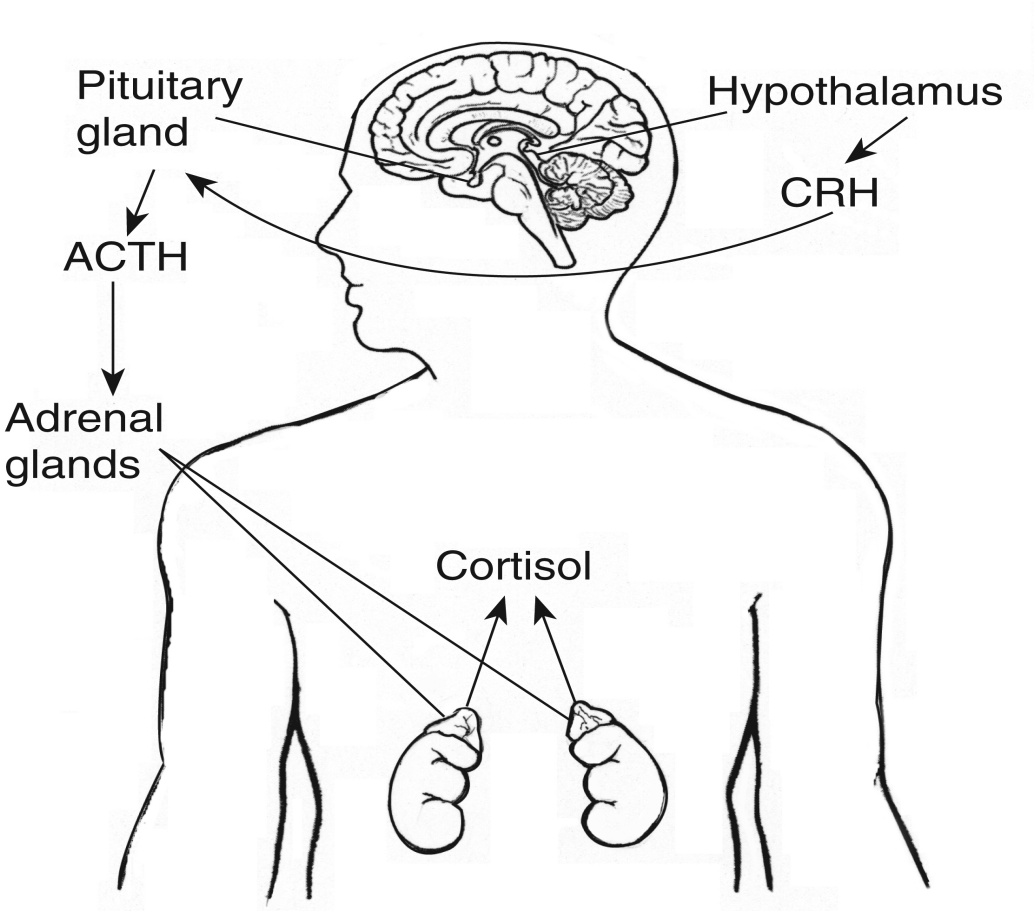


Figure 3.1: Adrenal glands

The adrenal cortex outer region is the largest part of an adrenal gland. It can be divided into three separate zones, which are the outer layer zona glomerulosa, the middle layer zona fasciculata, and the inner layer zona reticularis. Each zone is in charge of producing distinct hormones [65]. Each layer produces steroid hormones from the precursor cholesterol. Mineralocorticoids, glucocorticoids, and sex steroids (mostly DHEA with some androstenedione) are produced in the zona glomerulosa, zona fasciculata , and zona reticularis respectively.

The mineralocorticoids (the red pathway in Fig. 3.2) include deoxycorticosterone, cor-

ticosterone, and aldosterone, which are crucial in controlling blood pressure and certain electrolytes e.g. sodium and potassium. In particular, aldosterone is important in the whole process because it can act on the kidney to absorb more sodium into the blood-stream and excrete potassium into the urine. Thus, it helps to regulate the pH of the blood via controlling the levels of electrolytes in the blood [66]. Deoxycorticosterone and corticosterone also have mineralocorticoid effects but are much weaker than aldosterone.

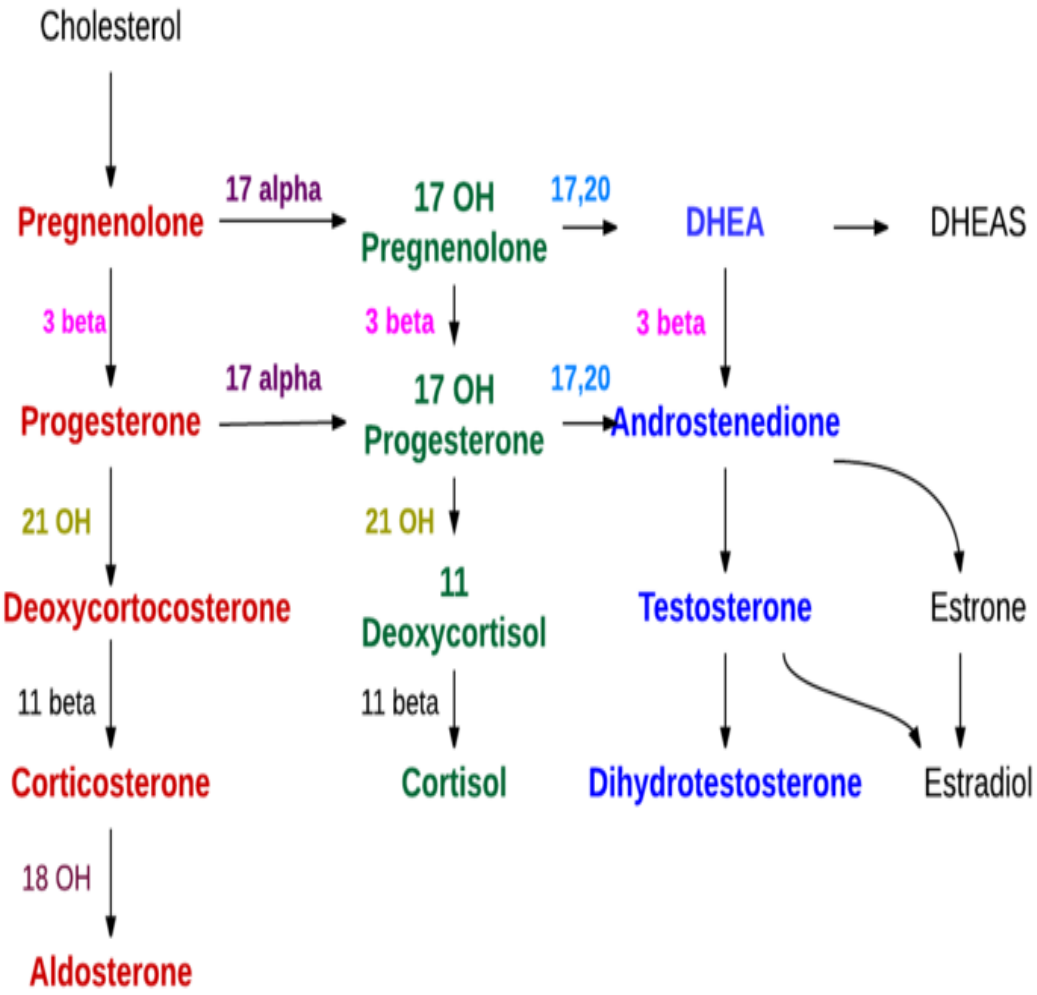


Figure 3.2: Adrenal steroid synthesis pathways

Cortisol, the major glucocorticoid (the green pathway in Fig. 3.2), plays several important roles in the body. It helps control the body's use of carbohydrates, proteins, and fats. It can also reduce inflammation, raise blood sugar, and regulate blood pressure. In addition, cortisol also controls the sleep/wake cycle and helps boost energy levels during times of stress [66].

Then, it is necessary to understand how this crucial hormone, cortisol is produced by adrenal glands. Generally, adrenal glands are responsible for producing hormones based on signals that are sent by the brain from the pituitary gland. It is the reaction triggered by the signals from the hypothalamus, which is also located in the brain. The whole process is called the hypothalamic pituitary adrenal axis. When it comes to producing cortisol, firstly, the hypothalamus secretes a hormone known as corticotropin-releasing hormone (CRH), which helps stimulate the pituitary gland to produce a hormone known as adrenocorticotropic hormone (ACTH). Then, the production and release of cortisol from the adrenal glands are stimulated by ACTH. Usually, the pituitary and the hypothalamus gland are able to sense if there is enough cortisol to meet the needs of the body. If there is an excess or deficiency of cortisol, both glands will alter the amount of the CRH and ACTH released. This situation is denoted as a negative feedback loop. There could be a variety of causes of the excess production of cortisol, such as tumors in the pituitary gland or nodules in the adrenal gland [65].

Cortisone is also a hormone secreted by the adrenal glands. It is a precursor of cortisol. However, cortisone is inactive in glucocorticoid activities [67]. Cortisol and cortisone can

interconvert with each other based on a certain ratio. Cortisol with the reaction of the 11bHSD-2 enzyme produces cortisone. Also, cortisone can be reactivated back to cortisol through the enzyme 11bHSD. Cortisone can cause blood pressure to rise in stressful situations as same as cortisol. It can also be used to treat adrenal insufficiency.

3.3 Adrenal gland disorders

Usually, hormonal imbalances can cause various health problems. The adrenal glands may produce too much or too little of certain hormones if some diseases exist in the adrenal glands or pituitary gland. There are many types of adrenal gland disorders depending on different hormones. Three disorders (Primary Aldosteronism, Cushing's syndrome, and adrenal insufficiency) are introduced in detail including their causes, symptoms, and diagnoses, etc.

3.3.1 Primary Aldosteronism

A condition known as Primary Aldosteronism occurs when the adrenal glands produce too much of the aldosterone [64]. It is discussed in the previous section that aldosterone is the main hormone to regulate blood pressure by balancing the levels of potassium and sodium in the body. Thus, the excess of aldosterone might cause high blood pressure and low potassium levels. The most straightforward way to diagnose Primary Aldosteronism is to test the aldosterone levels in the blood or urine, which help to determine the cause and confirm the diagnosis.

Primary Aldosteronism can be caused by the overactivity of both the adrenal glands [68]. A benign tumor on one of the adrenal glands is another cause of the condition. The overactivity of adrenal glands is caused by the overgrowth of adrenal tissue, which has effects on both adrenal glands and makes them generate too much aldosterone. It is not known why this overactivity occurs, but it accounts for around 60% to 70% of all cases of this disorder. Then, another 30% to 40% of people with Primary Aldosteronism have a benign tumor on one of their adrenal glands. In some rare cases, Primary Aldosteronism may occur as part of an inherited disorder.

One of the most common symptoms of Primary Aldosteronism is high blood pressure (even after taking blood pressure medicines). People who have this symptom may have adrenal glands issues, which causes the aldosterone excess. Then, the salt and water are retained by the body and raise the blood pressure. Another symptom is low potassium levels, while not all people with Primary Aldosteronism have it, but they might have fatigue, muscle cramps, and muscle weakness instead.

3.3.2 Cushing's Syndrome

Cushing's syndrome happens when there is too much cortisol in the body [69]. Clinically, a syndrome means that a group of symptoms and signs happen together. Cushing's syndrome is fairly rare and adults between 20 and 50 years old are affected mostly.

Cuhsing's syndrome can happen due to cortisol excess produced by the body itself (adrenal glands, pituitary gland, and hypothalamus control cortisol levels in the body). The un-

derlying causes of high levels of cortisol are listed below [65]:

- Adrenal cortical tumors. A tumor that grows on the adrenal gland can produce too much cortisol. Usually, it is benign. However, it can also be a rare adrenal cortical carcinoma.

- Pituitary tumors. Overproduction of ACTH caused by pituitary tumors, tells the adrenal glands to make cortisol. It leads to 8 out of every 10 cases of Cushing's syndrome. This type of Cushing's syndrome is known as Cushing's disease.

- Lung, pancreas, thyroid, and thymus tumors. Tumors outside of the pituitary gland can also produce ACTH. Usually, these tumors are malignant, for example, small cell cancer, which is the most common type.

Also, another main cause of Cushing's syndrome is the side effects of taking certain medications for a long time to treat some chronic diseases such as asthma.

Cortisol is an important hormone, which has a vector of effects on the body. Thus, Cushing's syndrome may have various symptoms. Some of these symptoms are unique to Cushing's syndrome but some could be related to other syndromes. Also, different people could have distinct symptoms. Thus, Cushing's syndrome is sometimes very hard to diagnose and mistaken for polycystic ovary syndrome or metabolic syndrome. In general, Cushing's syndrome can have symptoms of a red, round face, 'moon face ', 'buffalo hump', high blood pressure, excessive hair growth, diabetes, etc.

3.4 Summary

In conclusion, adrenal glands and their steroid pathways activities have great impacts on people's health. There are different hormones produced in each pathway. Cortisol and aldosterone are the two most important hormones that are glucocorticoid and mineralocorticoid respectively. If the cortisol and/or aldosterone are unbalanced in the body, people will have adrenal gland disorders, for example, Primary Aldosteronism, Cushing's syndrome, etc. All those disorders have to be treated in time, otherwise, serious symptoms like stroke, coma, etc. may occur.

Chapter 4 Unsupervised leaning - Self-organising maps in the model space

4.1 Introduction

In order to deal with and visualise sparsely sampled and noisy time series data, a novel SOM approach in the model space termed as SOM in the model space (SOMiMS) is provided, together with an extension of GTM to the model space. Those two are both LiMS approaches, which represent each time series by a model (point estimate) or a posterior distribution over models.

The SOMiMS is directly formulated in the model space by the probabilistic model formulation of the inferential mechanistic model, which keeps the classic SOM theory of retaining control over the neighbourhood-shrinking rate. The extended GTM has a clean formulation, but it is not able to manipulate the dynamic neighborhood size directly. Those two approaches are demonstrated on a real data set of measurements taken on subjects in an adrenal steroid hormone study.

The rest of the chapter is organised as: Section 4.2 proposes SOMiMS and extended GTM

models. Section 4.3 introduces the inferential mechanistic model and adrenal steroid hormone data set used by SOMiMS and extended GTM. Section 4.4 presents the experiment details and results. Then, this chapter is concluded in Section 4.5.

4.2 Methodologies

4.2.1 Topographic Mapping of time series in the model space

Consider an input data set of time series $\mathcal{Y} = \{Y_1, Y_2, ..., Y_N\}, n = 1, ..., N$. The n-th time series will be denoted by $Y_n = \{Y_n^t\}_{t=1,...,T_n}$, where T_n is the length of n-th time series. The length of each time series can be different, but observations of time series are permitted to be taken in a unique time grid.

In the LiMS approach, the model space refers to the inferential mechanistic model space. Each time series can be seen as the partial observations of some inferential mechanistic models parameterised by $\vec{\theta} \in \mathbb{R}^d$ [29]. The parametric mechanistic model is a system of multivariate ordinary differential equations (ODEs) mathematically.

Usually, once a vectorial data set is given, the topographic mapping is generated by a nonlinear mapping, which projects each data in the original space to a low-dimensional topographic mapping space [53][70]. Normally, such topographic mapping space is two-dimensional (for visualisation purposes). Topographic mappings that this research is interested in are going to work in the model space rather than the original signal space. Each node of the topographic mapping in the model space refers to an instance from the

underlying model class. Then, each time series is represented as an individual projection on the topographic mapping.

SOM in the model space - SOMiMS

Consider a SOM structured with $k \times k$ nodes. Each node notated as i is assigned by a parameter vector $\vec{\theta}_i$, which is representative of an inferential model. Given the n-th time series Y_n in the data set, the log-likelihood of i-th node is:

$$\mathcal{L}(Y_n|\vec{\theta}_i, \Sigma) = \ln \prod_{t=1}^{T_n} p(Y_n^t|\vec{\theta}_i, \Sigma) = \sum_{t=1}^{T_n} \ln p(Y_n^t|\vec{\theta}_i, \Sigma), \tag{4.1}$$

where Σ is a collection of parameters of the observational noise. As mentioned above the length of each time series may vary, so the log-likelihood in Eq. 4.1 is not comparable between time series. Thus, in order to have one log likelihood per observation, the log-likelihood is divided by the length of each time series T_n denoted by Q:

$$\mathcal{Q}(Y_n|\vec{\theta}_i, \Sigma) = \frac{1}{T_n}\mathcal{L}(Y_n|\vec{\theta}_i, \Sigma) = \frac{1}{T_n}\sum_{t=1}^{T_n} \ln p(Y_n^t|\vec{\theta}_i, \Sigma). \tag{4.2}$$

Then, given a time series Y_n, the $\overline{\mathcal{Q}}$ termed as "quality measure "of the i-th node is generated by renormalising through all nodes:

$$\overline{\mathcal{Q}}(Y_n|\vec{\theta}_i, \Sigma) = \frac{\mathcal{Q}(Y_n|\vec{\theta}_i, \Sigma)}{\sum_a \mathcal{Q}(Y_n|\vec{\theta}_a, \Sigma)} = \frac{-\mathcal{Q}(Y_n|\vec{\theta}_i, \Sigma)}{\sum_a -\mathcal{Q}(Y_n|\vec{\theta}_a, \Sigma)}. \tag{4.3}$$

The $-\mathcal{Q}(Y_n|\vec{\theta}_i, \Sigma)$ refers to the information (per observation) that the i-th node on the mapping contains about the given time series Y_n. Thus, the quality measure $\overline{\mathcal{Q}}$ renormalized across all nodes is the normalised information the node i has on Y_n.

The Gaussian observational model is employed. The formulation is given by:

$$p(Y_n^t|\vec{\theta}_i, \Sigma) = \frac{1}{(2\pi)^{\frac{D}{2}}|\Sigma|^{\frac{1}{2}}} \left\{ \exp(-\frac{1}{2}(Y_n^t - X_i^t)^{\mathbf{T}}\Sigma^{-1}(Y_n^t - X_i^t) \right\}, \tag{4.4}$$

where X_i^t refers to the vector of free noise observations at time t derived from the inferential model represented by $\vec{\theta}_i$. Here assume a homoscedastic process with a fixed covariance Σ, which is a collection of observational noise. Each data has an observational noise level estimated through Gaussian Process. The observational noise used in this homoscedastic process is obtained by averaging observational noise of all data.

In the training phase, a time series Y_n is randomly chosen from the data set $\mathcal{Y}$ with replacement in each iteration. Rather than updating the node with the maximum quality $\mathcal{Q}$, called winner node and its neighborhood like what classic topographic mapping does, every node is updated and its neighborhood based on $\overline{\mathcal{Q}}(Y_n|\vec{\theta}_i, \Sigma)$, the normalised quality measure. This is necessary as the data this research is interested in are sparsely observed and very noisy time series and one time series might be represented by a number of prototypical node models. The final topographic mapping result would be biased if the time series is only committed to a single winner node. Hence, each node should be considered in turn. Considering node i, its neighborhood nodes c will be updated as:

$$\vec{\theta}_c(l+1) = \vec{\theta}_c(l) \ + \ \overline{\mathcal{Q}}(Y_n|\vec{\theta}_i, \Sigma) \cdot h_{(c,i)}(l) \cdot \eta(l) \cdot \nabla_{\vec{\theta}_c} Q(Y_n|\vec{\theta}_c(l)), \tag{4.5}$$

in which $h_{(c,i)}(l)$ indicates the neighborhood function and $\eta(l)$ is the learning rate. Both

of them monotonically decrease during time steps l and are given by:

$$\eta(l) = \eta(0) \cdot \exp\left(-\frac{l}{\tau}\right), \tag{4.6}$$

$$h_{(c,i)}(l) = \exp\left\{-\frac{||c-i||^2}{2(\alpha(l))^2}\right\}, \tag{4.7}$$

$$\alpha(l) = \alpha(0) \cdot \exp\left(-\frac{l}{\tau}\right). \tag{4.8}$$

.

The τ in Eq. 4.8 is a time scale parameter and l is the index of the current iteration. $\eta(0)$ and $\alpha(0)$ are the initial learning rate in the power series learning rate function and the initial neighborhood size respectively.

To sum up, there are two significant aspects of the SOMiMS approach:

1. For each time series Y_n, a double update is conducted through the grid nodes
   - The outer update: through the pivotal nodes $\vec{\theta}_i$
   - the inner update: through their neighborhood $\vec{\theta}_c$
2. the updates of nodes are taken place in the model space through the gradient based algorithm learning in the directions ($\nabla_{\vec{\theta}_c} Q(Y_n|\vec{\theta}_c)$ of improving the node likelihoods once a time series Y_n is given.

In the end, it is important to make the SOMiMS able to be visualized in a two-dimensional map. Thus, the $k \times k$ node grid is embedded in a square of $[-1,1]^2$. Then, the embedded grid made up of points $\{\mathbf{g}_i\}_{i=1}^{J},\ \mathbf{g}_i \in [-1,1]^2$ is obtained. After that, the Y_n is visualized

in the $[-1,1]^2$ square as the mean of the posterior distribution over the grid points [21] [58]:

$$Proj(Y_n) = \sum_{i=1}^{J} P(\mathbf{g}_i|Y_n, \vec{\theta}_i, \Sigma) \cdot \mathbf{g}_i, \tag{4.9}$$

$$P(\mathbf{g}_i|Y_n, \vec{\theta}_i, \Sigma) = \frac{p(Y_n|\vec{\theta}_i, \Sigma)}{\sum_{j=1}^{J} p(Y_n|\vec{\theta}_j, \Sigma)}, \tag{4.10}$$

in which a uniform prior distribution over the grid is applied.

Generative Topographic Mapping in the model space

In the same lines with [59] and [58], the GTM [21] is also extended to the model space as an alternative to SOMiMS. Assume there is a two dimensional latent space $\mathcal{H} = [-1,1]^2$,which is covered by the regular grid $\{\mathbf{g}_i\}_{i=1}^{J}$ containing J points ($\mathbf{g}_i \in \mathcal{H}$). The objective is to represent each time series using the latent space $\mathcal{H}$ through imposing a uniform prior over the grid. The latent space is mapped into the model space via a function $\ell(\mathbf{g}; W)$ parametrised by W :

$$\ell(\mathbf{g}; W) = W\phi(\mathbf{g}), \tag{4.11}$$

in which W is a matrix in the shape of $d \times M$ that governs the mapping $\ell(\mathbf{g}; W)$ and $\phi(\mathbf{g})$ are fixed basis functions $\phi_m(\mathbf{g}), m = 1, ..., M : \mathcal{H} \to \mathbb{R}$. Compared to SOMiMS, now the $\ell(\mathbf{g}_i; W)$ plays the role of the $\vec{\theta}_i$, the i-th prototypical model representative.

Considering the n-th time series Y_n with the length T_n from the data set $\mathcal{Y}$, the likelihood of Y_n in the i-th parameter $\ell(\mathbf{g}_i; W)$ of the inferential forward ODE model is given by:

$$p(Y_n|\mathbf{g}_i, W, \Sigma) = \prod_{t=1}^{T_n} p(Y_n^t|\ell(\mathbf{g}_i : W), \Sigma), \tag{4.12}$$

where Σ is the collection of observational noise model parameters.

The data log-likelihood can be formed by:

$$\mathcal{L} = \sum_{n=1}^{N} \ln \left\{ \frac{1}{J} \sum_{i=1}^{J} p(Y_n|\mathbf{g}_i, W, \Sigma) \right\}, \tag{4.13}$$

as GTM is a flat mixture model of the latent grid. Then, the Expectation Maximization (EM) algorithm is employed to get parameter W by maximizing $\mathcal{L}$. Given time series Y_n, the R_{in} refers to the 'responsibilities 'of grid points $\mathbf{g}_i, i = 1, ..., J$ are formulated in the E-step as:

$$R_{in} = p(\mathbf{g}_i|Y_n, W, \Sigma) = \frac{p(Y_n|\mathbf{g}_i, W, \Sigma)}{\sum_j p(Y_n|\mathbf{g}_j, W, \Sigma)}. \tag{4.14}$$

Then, the expected complete-data log-likelihood is given by:

$$\langle \mathcal{L}_{comp} \rangle = \sum_{n=1}^{N} \sum_{j=1}^{J} R_{in} \ln\{p(Y_n|\mathbf{g}_i, W, \Sigma)\}. \tag{4.15}$$

In the end, M-step is used to maximise $\langle \mathcal{L}_{comp} \rangle$ respecting to W.

To sum up, there are four main steps of the GTM in the model space approach:

1. Map the latent space $\mathcal{H}$ into the model space via $\ell(\mathbf{g}; W)$ parametrised by W

2. The probability distribution over the model space constrained by the latent space $\mathcal{H}$ is obtained by $p(Y_n|W, \Sigma)$

3. The data log-likelihood is formed by $\mathcal{L}$ in Eq. 4.13

4. Use Expectation Maximization algorithm to get parameter W by maximizing $\mathcal{L}$.

Once the training finishes and optimal W is obtained, each time series can be visualised in the latent space $\mathcal{H}$ as Y_n will be transferred to the mean of the posterior distribution over grid points in the latent space [21] [58]:

$$Proj(Y_n) = \sum_{i=1}^{J} R_{in} \cdot \mathbf{g}_i. \tag{4.16}$$

4.3 Inferential biomedical model

Both SOMiMS and the extension of GTM are demonstrated on the real-world adrenal steroid hormone data set. Major adrenal steroid hormones are produced by different areas of the adrenal cortex: glucocorticoids, mineralocorticoids, and sex steroids [71]. In this study, glucocorticoids and mineralocorticoids are focused on particularly. An exploration of these pathways helps to understand the different forms of congenital adrenal hyperplasia (CAH) and isolated hypoaldos-teronism characterised by defects in the functionality of enzymes involved in adrenal steroid hormone synthesis [72].

In this chapter, an inferential mechanistic ODE model subjecting to upstream circadian regulation is used. The model contains four hormones (Corticosterone, Aldosterone, Cortisol, and Cortisone), which are representatives of both the glucocorticoid and mineralocorticoid pathways (Fig.4.1).

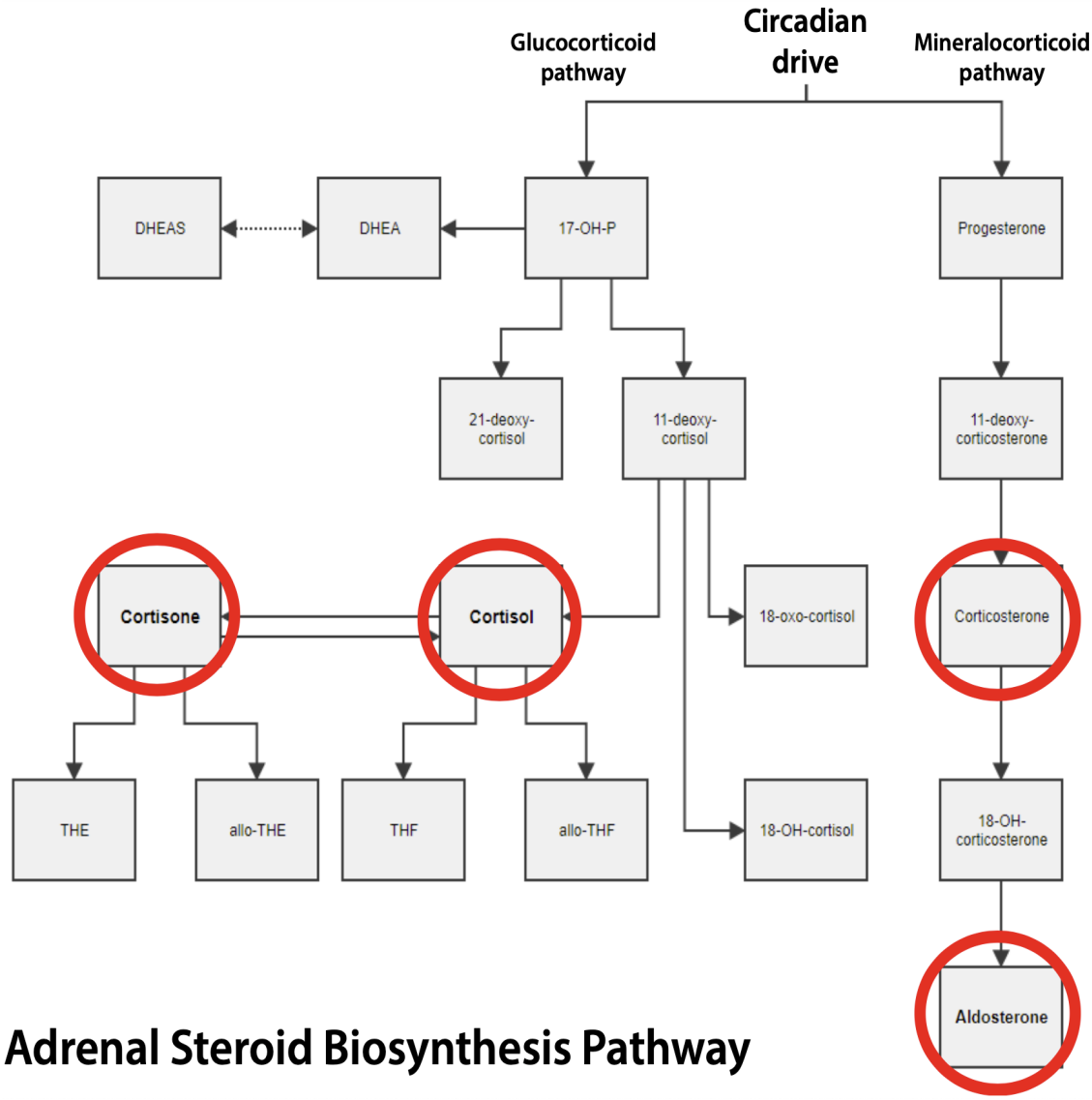


Figure 4.1: Adrenal Steroid Biosynthesis Pathway. *Left branch: glucocorticoid pathway. Right branch: mineralocorticoid pathway. Modelled hormones are circled in red.*

The system of coupled ODEs is given by:

$$\frac{d}{dt}C = K_C\varphi_c(t) - K_A C - \gamma_C C, \tag{4.17}$$

$$\frac{d}{dt}A = K_A C\varphi_u(t) - \gamma_A A, \tag{4.18}$$

$$\frac{d}{dt}F = K_F\varphi_c(t) - K_E F + K_b E - \gamma_F F, \tag{4.19}$$

$$\frac{d}{dt}E = K_E F - K_b E - \gamma_E E, \tag{4.20}$$

where C, A, E, and F corresponds to Corticosterone, Aldosterone, Cortisol, and Cortisone, and $\varphi_c(t)$ and $\varphi_u(t)$ are periodic circadian and ultradian drives specified by:

$$\varphi_c(t) = \alpha_c \sin(2\pi(t + T_s^c) + \sigma \sin(2\pi(t + T_s^c))) + \beta,$$

$$\varphi_u(t) = 1 + \alpha_u \sin(2\pi(t + T_s^u)n_p).$$

Typically, CAFE hormones are circadian rhythmic so the asymmetrical circadian drive is modelled by $\varphi_u(t)$. The $\varphi_u(t)$ only acts on Aldosterone synthesis.

In total, there are sixteen parameters used in mechanistic models, whose descriptions are listed in Table 4.1. After consultation with our biomedical collaborators, three drive parameters $\alpha_c = 1$, $\alpha_u = 1$, $T_s^u = 0.5$ are fixed, leaving thirteen free parameters.

The real data set used in this chapter corresponding to the inferential mechanistic ODE models described above includes three conditions: Healthy control, Cushing's, and Primary Aldosteronism (PrimAldo). Usually, the excessive production of Cortisol results in the Cushing's. The Aldosterone excess is the cause of PrimAldo. The data contains 60 subject-specific multivariate time series of Corticosterone, Aldosterone, Cortisol, and Cortisone. Those 60 subjects covering the three conditions include 30 Control, 15 Cushing's, and 15 PrimAldo. Each time series is sampled every twenty minutes within twenty-four hours. However, the length of each sampled time series may vary because there are some missing values due to operational issues.

Table 4.1: Model parameters for clustering

Parameter	Description	Parameter	Description
K_C	Corticosterone synthesis rate	K_A	Aldosterone synthesis rate
K_F	Cortisol synthesis rate	K_E	Cortisone synthesis rate
K_b	Cortisone to Cortisol conversion rate	γ_C	Corticosterone degradation
γ_A	Aldosterone degradation rate	γ_F	Cortisol degradation rate
γ_E	Cortisone degradation rate	α_c	Amplitude of circadian drive
T_s^c	Phase shift of circadian drive	σ	Asymmetry of circadian drive
β	Offset of circadian drive	α_u	Amplitude of ultradian drive
T_s^u	Phase shift of ultradian drive	n_p	Number of ultradian pulses

4.4 Experiments and results

This section delivers the experiment details and results of SOMiMS and the extension of GTM methodologies applied to the real adrenal steroid dataset.

To initialise both SOMiMS and extended GTM, a 10×10 grid is adopted and the models are initialized based on the classic SOM in the signal space trained on all 60 subjects. Note that in the signal space, each time series may have a different length because of missing values. The univariate Gaussian process (GP) model [73] is used to impute the missing values. Once the training is completed, most grid points of the classic SOM are assigned one or more time series. Also, there might be some grid points with no time series assigned. Under these circumstances, time series assigned to their closest neighbours

are used to represent them. In the end, each grid point has its corresponding time series. Then, those grid points are transferred to the model space by calculating the maximum likelihood estimations on parameters given the time series assign to each of them. Consequently, the 10×10 classic SOM map is transferred to a map in the model space, in which each grid point refers to a set of a vector containing thirteen parameters.

The classic SOM in the signal space, which is the initialization for SOMiMS and extended GTM is trained for 300 epochs. The initial learning rate and initial neighbourhood size for the classic SOM are set as 0.2 and 6. As the classic SOM already captured the rough initial topographical organisation structure, the SOMiMS initialized by classic SOM is trained just for 200 epochs with the initial learning rate and initial neighbourhood size of 0.1 and 2 respectively. In the extension of GTM, M=16 (4×4) basis functions ϕ_m and one additional constant basis function as the bias term are used. The Gaussian functions were employed for basis functions with the same width of $\sigma = 1$. After running 80-100 E-M cycles, the likelihood stepped up.

Finally, topographic maps generated by SOMiMS and extended GTM are illustrated in Fig 4.4. Both methodologies are trained in an unsupervised manner, which means labels of the data projections corresponding to their conditions were not used during the whole training process. As a whole, both topographic maps in the model space constructed by SOMiMS and extended GTM reveal a good degree of separation of Control (blue circle), Cushing's (red square), and PrimAldo (green triangle), especially this is a noisy and sparse data set measured on real subjects and containing missing values. In addition, both maps

show the sub-grouping structure of Cushing's cohort into at least sub-populations. Two plots (each contains four subplots) on both sides of the SOMiMS map (Fig.4.3) are trajectories of steroid time series corresponding to two selected subjects in the SOMiMS map.

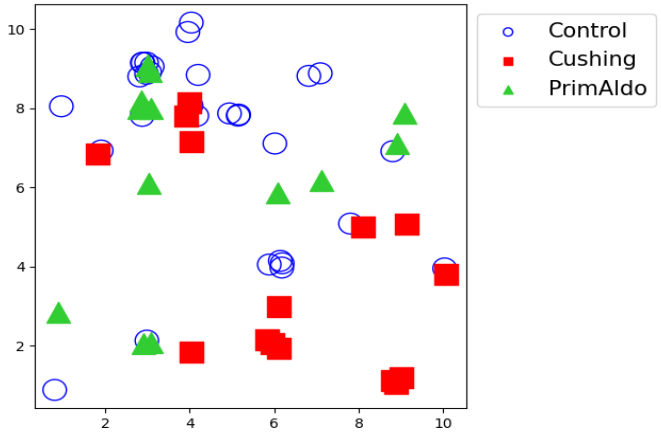


Figure 4.2: Topographic visualization of the data obtained by Classic SOM *Initialisation of SOMiMS and Extended GTM*

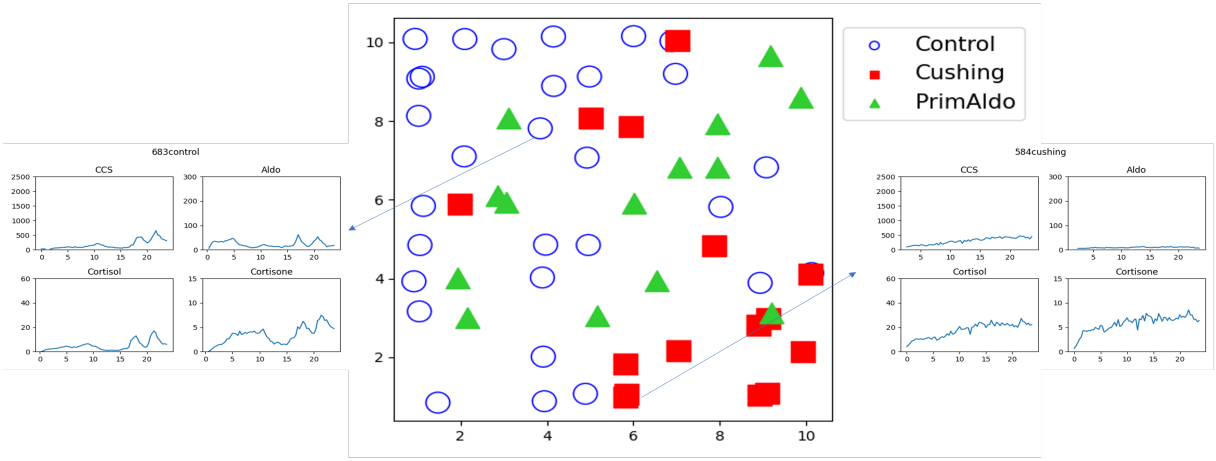


Figure 4.3: Topographic visualization of the data obtained by SOMiMS

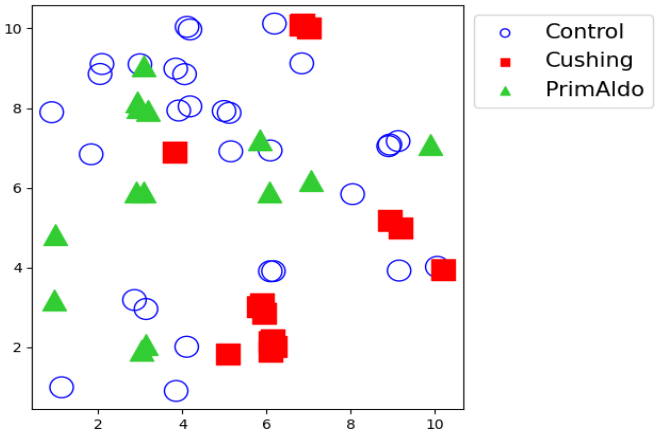


Figure 4.4: Topographic visualization of the data obtained by Extended GTM

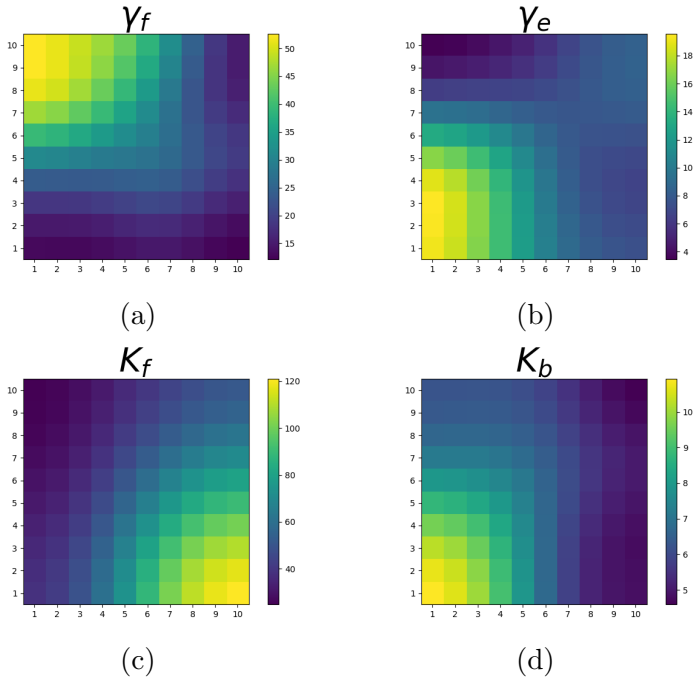


Figure 4.5: Parameter heat maps of γ_F (a) , γ_E (b), K_f (c) and K_b (d) for the SOMiMS model.

One of the dominating advantages of our topographic mapping methodologies in the model space is the chance to promptly interpret the topographic structure and data organisation from the mechanistic point of view of the underlying processes that how data are generated. The underlying mechanistic information can be obtained by generating the parameter plots, which are heat maps and show the values learnt of each individual

parameter in the mechanistic model across the prototypes on the 10×10 grid. In Fig. 4.5, there are two parameter plots for Cortisol and Cortisone degradation rates (γ_F and γ_E). Clearly, both heat maps have low values of γ_F and γ_E in the areas of SOMiMS topographic map which contain Cushing's projections (in the lower right corner). It is the case that Cushing's condition is related to the Cortisol excess, which is partially caused by the low degradation rate of Cortisol. Cortisone is positively coupled with Cortisol by K_b so the low value of γ_E can also lead to Cortisol excess. The K_f and K_b heat maps have similar interpretations. The parameter plot for every individual model parameter can be plotted.

Table 4.2: SOMiMS KNN confusion matrix

SOMiMS	True Control	True Cushing's	True PrimAldo
Predicted Control	0.80	0.10	0.10
Predicted Cushing's	0.13	0.67	0.20
Predicted PrimAldo	0.21	0.29	0.50
Sensitivity	0.70	0.63	0.65
Accuracy	0.82	0.76	0.73

Table 4.3: Extended GTM KNN confusion matrix

SOMiMS	True Control	True Cushing's	True PrimAldo
Predicted Control	0.68	0.27	0.05
Predicted Cushing's	0.13	0.80	0.07
Predicted PrimAldo	0.14	0.22	0.64
Sensitivity	0.72	0.62	0.66
Accuracy	0.80	0.82	0.84

In addition, the KNN [74] is performed in order to qualify the degree of separation of these three conditions on the topographic mapping (visualisation plot). $K = 3$ is picked after the cross-validated hyperparameter tuning. Table 4.2 and 4.3 are KNN confusion matrices of SOMiMS and extended GTM respectively. In general, the topographic mapping organisation is not directly related to the classification performance. The projection structure is more spread of SOMiMS than the extension of GTM, which is better for visualisation. This might benefit from the possibility of explicit control over the topographic map formation offered by SOMiMS (remembering the neighbourhood function and its shrinkage of SOMiMS). Also, the methodologies presented in this chapter are under the unsupervised learning scenario, in which the full formation of a topographic map may disrupt cases of multiple projections in a very close neighbourhood of the visualisation space. However, a scenario that could yield good distance-based classifications are not preferable from the visualisation point of view.

4.5 Summary

In this chapter, a new SOM approach, SOMiMS has been proposed, which is formulated directly in the model space. An extended GTM formulation has also been generated in the model space. Both two approaches are able to visualize sets of sparse and/or noisy time series on a two-dimensional map. These two methodologies have been demonstrated on a real data set of measurements on subjects with different steroid hormone biosynthesis conditions (Control, Cushing' and PrimAldo). In addition, a mechanistic inferential model has been formulated with thirteen free parameters in the format of coupled ordinary differential equations. How the topographic maps could be formed in the space of these coupled models given the data had been introduced. Parameter plots (heat maps) based on topographic maps to interpret the topographic structure from the mechanistic point of view have also been provided.

Consequently, compared to classic SOM and GTM, our SOMiMS and extended GTM are not only able to deal with sparse and/or noisy time series with missing values but also can create interpretable readily data visualisations and take the mechanistic information into account.

Chapter 5 Supervised learning - Classification in the model space

5.1 Introduction

In this chapter, the time series classification in the model space applied to the adrenal steroid hormone data set is presented. Compared to the signal space, classification in the model space is not only naturally able to handle sparse and/or noisy time series data with significant performance but is also capable of taking mechanistic and biomedical knowledge into account. It can help to understand what mechanistic parameters should be focused on. Besides the classification in the model space, the classification in the signal space is also included as a benchmark. Additionally, a hybrid classification model combining the classification models in the signal and model space is developed with satisfactory performance. Classification and feature selection results, together with point estimations plots of important parameters are presented and interpreted from the mechanistic and biomedical points of view. Moreover, the classification in the model space is also demonstrated on an unfiltered data set, which contains more data subjects and larger observational gaps.

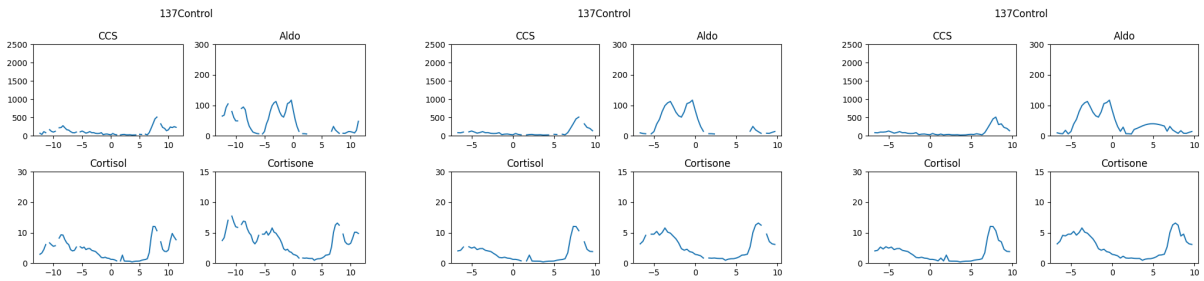
The rest of the chapter is organised as follows: Section 5.2 is the explanation of the data form in both signal and model space, along with the Gaussian process used to impute missing values and the inferential biomedical model. Section 5.3 introduces two classification models (SVM and logistic regression) used in this task, following with the solution to the class imbalance problem. It also includes an explanation of the feature selection based on importance plots. Section 5.4 is the experimental methodology including the experiment design based on 3 degrees of design freedom and some hyperparameter tuning, training, and validation details. Results and discussion of the filtered data set are presented in Section 5.5. Section 5.6 involves some additional work for the unfiltered data set and its corresponding results. Then, this chapter is concluded in Section 5.7.

5.2 The data and inferential biomedical model description

5.2.1 Data in the signal space

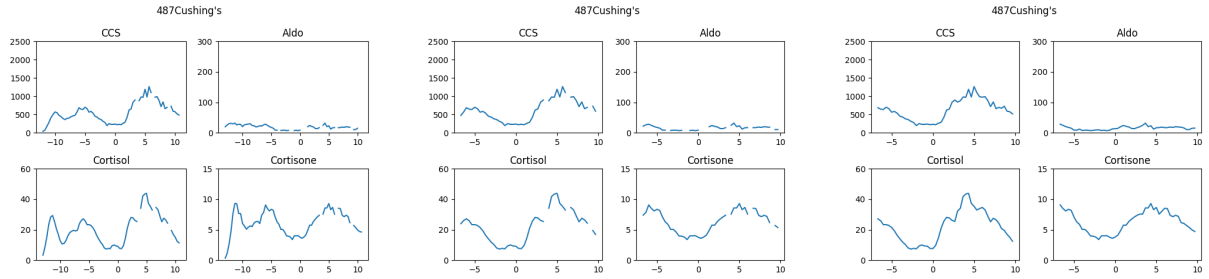
The adrenal steroid hormone data set with 140 subjects contains three conditions: Control, Cushing's, and PrimAldo. Those three classes are imbalanced with 100 Control, 22 Cushing's, and 18 PrimAldo. Each individual data is a multivariate time series with four metabolites: Corticosterone (C), Aldosterone (A), Cortisol (F), and Cortisone (E). All time series are observed every 20 minutes for a 24-hour period with different starting times. However, each metabolite has missing values at random time points so the length of the data can vary. In order to conduct classification in the signal space, all time series have to have the same length. For example, the K-nearest neighbor classifier (KNN) re-

quires the distance measure between time series pairs. Euclidean distance is widely used for time series classification [20]. However, Euclidean distance is not applicable for time series with variable lengths. Hence, the univariate Gaussian process (discussed in the next section) is adopted to impute those missing values. Then, all time series can be considered as feature vectors for classification. However, the starting time of each measurement process varies across the data items. This misalignment of measurement times could hamper classification results because the metabolites' level is time-dependent. More importantly, the times are treated as unique feature variables and their importance is evaluated with respect to the classification performance. Thus, all time series data were pre-processed so that the first measurement is taken at 5 pm and the last one is taken at 10 am the next morning. Figs 5.1, 5.2 and 5.3 show trajectories of three data subjects (Control 137, Cushing's 487, and PrimAldo 479), including their raw trajectory, trajectories after the alignment (5 pm - 10 am) and with the imputation of missing values. The 0 in the horizontal axis refers to midnight.



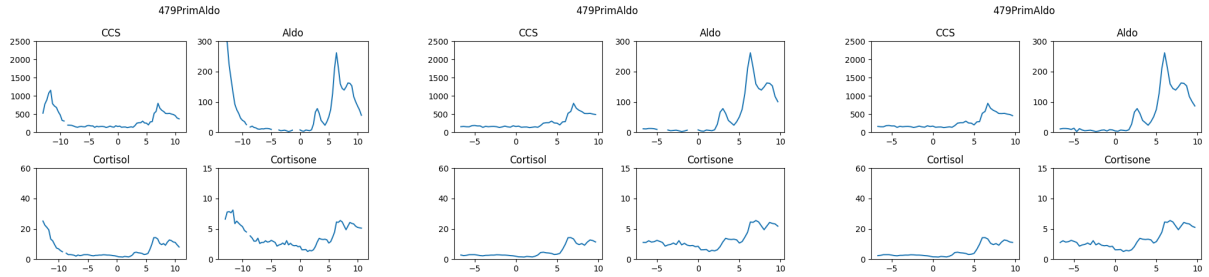
(a) raw trajectory (b) after alignment (c) with imputation

Figure 5.1: Trajectories of raw data 5.1a, after alignment5.1b and with imputation5.1c of Control 137



(a) raw trajectory (b) after alignment (c) with imputation

Figure 5.2: Trajectories of raw data 5.2a, after alignment5.2b and with imputation5.2c of Cushing's 487



(a) raw trajectory (b) after alignment (c) with imputation

Figure 5.3: Trajectories of raw data 5.3a, after alignment5.3b and with imputation5.3c of PrimAldo 479

5.2.2 Univariate Gaussian Process to fill in missing value

Supervised learning such as classification for discrete outputs and regression for continuous outputs is a crucial constituent of machine learning and statistics, either for data analysis or as a subgoal of a complex problem. Normally, parametric models are used for this purpose because they are easy to interpret. However, such simple parametric models lack expressive power for complex data [75]. Additionally, it is not easy to work with their complex structures in practice for example feed-forward neural networks. Hence, flexible models such as Gaussian processes (GP) that are practical to work with have been developed.

A stochastic process contains a collection of random variables indexed by the input vector $\boldsymbol{t}$ [76]. There are infinite random variables in the collection, while only some of them are observed finitely $t_1, ..., t_n$, which are training data points. The process is specified by defining the joint probability density that any finite number of random variables follows. Hence, a GP is a stochastic process in which every joint density function is Gaussian so it is fully specified by its mean and covariance [73]. Compared to multivariate Gaussian distribution, GP is over functions rather than vectors. In the univariate GP, random variables are related to a single process denoted as $\boldsymbol{f}$. It can be written as:

$$\boldsymbol{f} \sim \mathcal{GP}(m, k), \tag{5.1}$$

meaning that the function $\boldsymbol{f}$ is assumed to follow a GP with mean function m and covariance/kernel function k. For simplicity, GP is assumed to have zero means in practice. Then, all required to be defined is the covariance between two points t_i and t_j denoted as $k(t_i, t_j)$. Normally, the covariance function is specified by the squared exponential function as:

$$k(t_i, t_j) = \sigma^2 \exp\left(-\frac{1}{2l^2}(t_i - t_j)^2\right) + \delta_{ij}\sigma^2_{noise}, \tag{5.2}$$

where $\sigma^2 > 0$ is a scaling factor, which determines the variation of the GP from their mean. A small σ^2 means that the process stays close to its mean value and large σ^2 allows more variation. $l > 0$ is the lengthscale and it determines the smoothness of the GP. The square exponential term captures the concept that if t_i and t_j are close in the input space, their corresponding outputs have to be highly correlated. Moreover, σ^2_{noise} is the

noise variance, which is applied only when $i = j$. The noise variance is not a formal part of the covariance function. It is used by the GP to allow certain noise in the training data.

The GP defined above is prior to Bayesian inference so it just specifies some properties of the functions and is not related to any training data. The goal of the GP model is to make a prediction once a test point comes. Consider a training data set $D = (t_1, x_1), ..., (t_n, x_n), n = 1, ..., N$, where x_n is a sample from a random variable $f(t_n)$. Consider a new test input t^*, and then f^* is required to be predicted by computing the conditional distribution $p(f^*|f_1,..., f_N)$. This distribution is Gaussian determined by its mean and variance because the model used here is a GP, and there are standard formulae to calculate these values for a Gaussian distribution. Then $\boldsymbol{K}_+$, the covariance matrix of $(f_1, f_2, ..., f_N, f^*)$ can be partitioned into:

$$\boldsymbol{K}_+ = \begin{bmatrix} \boldsymbol{K} & \boldsymbol{k} \\ \boldsymbol{k}^T & k^* \end{bmatrix}, \tag{5.3}$$

where $\boldsymbol{K}$ is the covariance matrix of the training data, k^* denotes the variance of t^* and $\boldsymbol{k}$ $(N \times 1)$ is the covariance between the training data and t^*. Thus, the predicted mean and covariance of the test data t^* are given by:

$$\boldsymbol{E}[f^*] = \boldsymbol{k}^T \boldsymbol{K}^{-1} \boldsymbol{x}, \tag{5.4}$$

$$\boldsymbol{var}[f^*] = k^* - \boldsymbol{k}^T \boldsymbol{K}^{-1} \boldsymbol{k}. \tag{5.5}$$

Therefore, it is quite straightforward to make predictions on new test points once a covariance function is fixed. However, it is not realistic that there is enough information

about the parameters of a covariance function. An obvious approach is to adapt all parameters to model a training data set. Usually, this can be done by optimisation using the maximum likelihood framework. In order to do this, the log marginal likelihood of the training data given the hyperparameters should be computed. Then, a point estimate of all these hyperparameters is computed using gradient-based optimization.

After the GP model parameters are learned, the fitted model can be used to predict the f^* at index t^*, where the missing value x^* is a sample from the random variable f^*. In practice, the missing value x^* is filled by the mean of f^*. Note that each metabolite time series is normalized to zero-mean and unit variance before feeding it to model training and re-scale back all missing values computed during the prediction phase. In practice, this makes the training process rather robust when training GP models from a large set of time series data.

5.2.3 Inferential biomedical model

The joint inferential mechanistic ODE model below still represents the biosynthesis of adrenal steroid hormones subject to upstream circadian regulation. Four hormones, Corticosterone(C), Aldosterone(A), Cortisol(F), and Cortisone(E) are modelled in the first instance. They are representatives of both the glucocorticoid (F&E) and mineralocorticoid (C&A) pathways. Compared to the model in Chapter 4, the model here is updated by adding κ_c and κ_f, which are coefficients of convex combinations in Corticosterone and Cortisol respectively for interpretation purposes (in particular for Cushing's condition) after consultation with biomedical collaborators. The model is given by:

$$\frac{dC}{dt} = K_C \cdot \varphi_c^C(t) - \gamma_C \cdot C, \tag{5.6}$$

$$\frac{dA}{dt} = K_A \cdot C \cdot \varphi_u(t) - \gamma_A \cdot A, \tag{5.7}$$

$$\frac{dF}{dt} = K_F \cdot \varphi_c^F(t) + K_b \cdot E - \gamma_F \cdot F, \tag{5.8}$$

$$\frac{dE}{dt} = K_E \cdot F - K_b \cdot E, \tag{5.9}$$

where $\varphi_c^C(t)$ and $\varphi_c^F(t)$ are convex combinations of circadian drives and constant in-flow terms. The coefficients of the convex combinations are κ_c, $1-\kappa_c$, $\kappa_c \in (0,1)$ and κ_f, $1-\kappa_f$, $\kappa_f \in (0,1)$, for C and F, respectively:

$$\varphi_c^C(t) = \kappa_c \cdot \frac{1}{2} \cdot (sin(2\pi(t+T_s^c) + \sigma \cdot sin(2\pi(t+T_s^c))) + 1) + 1 - \kappa_c, \tag{5.10}$$

$$\varphi_c^F(t) = \kappa_f \cdot \frac{1}{2} \cdot (sin(2\pi(t+T_s^c) + \sigma \cdot sin(2\pi(t+T_s^c))) + 1) + 1 - \kappa_f, \tag{5.11}$$

The parameters $\kappa_c, \kappa_f \in (0,1)$ are parametrized through logistic function:

$$\kappa_c = \frac{1}{1+e^{-a_c}}, \tag{5.12}$$

$$\kappa_f = \frac{1}{1+e^{-a_f}}, \tag{5.13}$$

using unconstrained real-valued parameters a_c, a_f.

The ultradian drive $\varphi_u(t)$ only applied to Aldosterone(A) is captured through:

$$\varphi_u(t) = 1 + sin(2\pi(t+T_s^u) \cdot n_p). \tag{5.14}$$

All fourteen parameters used in our models and their descriptions are listed in Table 5.2. After discussing with medical and modelling collaborators, the phase shift of ultradian drive T_s^u is fixed to 0.5, leaving thirteen free parameters.

5.2.4 Data representation in the model space

Leaning in the model space is different from the signal space, in which input data are vectors of model parameters. In sub-section 5.2.3, the inferential biomedical model has a set of free parameters denoted as $\vec{\boldsymbol{\theta}}$. Consider a time series $\boldsymbol{x}_i$ represented by $\vec{\boldsymbol{\theta}}_i \in R^d$, where d is the dimensionality of parameters. The $\vec{\boldsymbol{\theta}}_i$ is obtained by running constrained maximum likelihood estimation [77] on the inferential mechanistic model.

Imaging the i-th time series $\boldsymbol{x}_i = \{\boldsymbol{x}_i^t\}_{t=1,\ldots,T_i}$, the principle of maximum likelihood estimation yields a choice of the estimator $\vec{\boldsymbol{\theta}}_i$ as the set of model parameters that make $\boldsymbol{x}_i$ most probable. The log-likelihood is formulated as:

$$\mathcal{L}(\boldsymbol{x}_i|\vec{\boldsymbol{\theta}}, \Sigma) = \ln \prod_{t=1}^{T_i} p(\boldsymbol{x}_i^t|\vec{\boldsymbol{\theta}}, \Sigma) = \sum_{t=1}^{T_i} \ln p(\boldsymbol{x}_i^t|\vec{\boldsymbol{\theta}}, \Sigma), \tag{5.15}$$

where Σ collects parameters of the observational noise. However, each individual parameter has to have a boundary to ensure they are biologically meaningful. After discussion with biomedical collaborators, the restrained parameter space is denoted as Ω. The ranges of parameters are listed in Table 5.2. Then, the maximum likelihood estimator is obtained through:

$$\vec{\boldsymbol{\theta}}_i = \arg\max_{\vec{\boldsymbol{\theta}}} \mathcal{L}(\boldsymbol{x}_i|\vec{\boldsymbol{\theta}}, \Sigma), \ \forall\ \vec{\boldsymbol{\theta}}_i \in \Omega, \tag{5.16}$$

The gradient-based optimization is used here to get the maximum likelihood estimator $\vec{\boldsymbol{\theta}}_i$. Initial values $\vec{\boldsymbol{\theta}}_0$ are given in Table 5.1. These initial values of parameters are obtained by fitting to rolling window averages of all 140 data subjects.

K_C	K_A	γ_C	γ_A	K_F	K_E	K_b	γ_F	a_C	a_F	σ	T_s^c	n_p
6000	1.22	100	7.47	17.78	9.78	2.34	10	0	0	1.09	0.43	4

Table 5.1: Initial values of model parameters

The step size used for gradient updating is 0.001. Other step values (0.1, 0.01, and 0.0001) are also tried. The optimisation is iterated 2500 times. Then constrained MLE is applied to each $\boldsymbol{x}_i$ and the input data set in the model space is obtained.

The constrained MLE is applied to both imputed data and raw data (with missing values). Both results will be presented. Thus, the classification in the model space is able to handle data sets with missing values compared to the signal space, in which every time series has to have the same length (with no missing values) and the same starting time. Moreover, the mechanistic model which is coupled ODEs described in section 5.2.3 is required to be solved to compute the log-likelihood. Values at the time point where four metabolises all have measurements are used as the initial values to solve ODEs.

Table 5.2: Model parameters for classification

Parameter	Description	Boundary
K_C	Corticosterone synthesis rate	1000 - 100000
K_A	Aldosterone synthesis rate	0.02 - 20
K_F	Cortisol synthesis rate	5 - 500
K_E	Cortisone synthesis rate	1 - 100
K_b	Cortisone to Cortisol conversion rate	0.3 - 30
γ_C	Corticosterone degradation	10 - 1000
γ_A	Aldosterone degradation rate	0.8 - 80
γ_F	Cortisol degradation rate	1 - 100
T_s^c	Phase shift of circadian drive	0.01 - 0.9
σ	Asymmetry of circadian drive	0.01- 5
κ_c	Coefficient of the convex combinations in Corticosterone	0 - 1
κ_f	Coefficients of the convex combinations in Cortisol	0 - 1
a_C	Logistic function parameter	-4 - 4
a_F	Logistic function parameters	-4 - 4
T_s^u	Phase shift of ultradian drive	0.5
n_p	Number of ultradian pulses	2 - 8

5.3 Classifier description

Although the adrenal steroid hormone data set contains three conditions, binary classifiers are used here for Control versus Cushing's and Control versus PrimAldo. Clinically it is more important to distinguish disease conditions (Cushing's or PrimAldo) from Control rather than classify disease conditions. In addition, repeated undersampling and classifier ensembles are adopted to address the class imbalance issue in three conditions. Moreover, a robust selection of important features is developed based on the coefficients of classifier models.

5.3.1 Classification models

Two classification models are employed to ensure consistent outcomes. They are linear SVM and logistic regression. There are two reasons why we choose these two classifiers. Firstly, these two models are based on different principles. Comparing two models under a same principle is not able to ensure consistent results. Secondly, they both have the coefficient vector $\boldsymbol{w}$, which plays an significant role in the feature selection algorithm introduced in section 5.3.3.

Support Vector Machine

SVM is a well-known learning method for binary classification. The general idea of SVM is to find a hyperplane that can separate the d-dimensional data into two classes by solving a constrained quadratic optimization problem [78][79]. However, for data that are not linearly separable, different kernel functions can be included in the model. It casts data

into a higher space where the data are separable[80] [79]. Here, the linear SVM is utilised because the data are linearly separable.

Consider l training examples $\{\boldsymbol{x}_i, y_i\}$, $i = 1, ..., l$ are given, where $\boldsymbol{x}_i \in R^d$ and $y_i \in \{-1, 1\}$. The input $\boldsymbol{x}_i$ can have d dimensions and y_i is the label. Then the aim is to construct a separating hyperplane (decision boundary) to separate positive subjects from negative ones with a maximum margin. It can be formulated as the constrained optimisation problem below:

$$\min \ \frac{1}{2}||\boldsymbol{w}||^2 + C\sum_{i=1}^{l}\xi_i, \tag{5.17}$$

$$\boldsymbol{s.\ t.} \quad y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i + b) \geq 1 - \xi_i,\ \xi_i \geq 0,\ i = 1, \ldots, l, \tag{5.18}$$

where $||\boldsymbol{w}||$ refers to the Euclidean norm of $\boldsymbol{w}$. C is a hyperparameter and it is greater than 0. The ξ_i are non-negative slack variables, which are used to ease the constraints and allow some misclassification. According to Karush-Kuhn-Tucker (KKT) Conditions [81], this optimization problem is normally transformed into its dual problem:

$$\max_{\alpha} \ \sum_{i=1}^{l}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{l}\alpha_i\alpha_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j, \tag{5.19}$$

$$\boldsymbol{s.\ t.} \quad \sum_{i=1}^{l}\alpha_i y_i = 0,\ 0 \leq \alpha_i \leq C,\ i = 1, \ldots, l.$$

Once optimal α are obtained, the decision function for a new input vector $\boldsymbol{x}_*$ is formulated as:

$$F(\boldsymbol{x}_*) = sign\left(\sum_{i=1}^{l}\alpha_i y_i \boldsymbol{x}_i^T\boldsymbol{x}_* + b\right), \tag{5.20}$$

where $\boldsymbol{w} = \sum_{i=1}^{l} \alpha_i y_i \boldsymbol{x}_i$.

Logistic regression

The logistic regression model is a type of statistical regression analysis technology. According to the number of values of the dependent variable, the logistic regression model can be categorised into binomial and multinominal regression[82]. Binary logistic regression is used in this work so the dependent variable y is normally coded as "0" and "1" representing two classes. Given the training data $\{\boldsymbol{x}_i, y_i\}$, $i = 1, ..., l$, where $\boldsymbol{x}_i \in R^d$ and $y_i \in \{0, 1\}$, the logistic regression takes the following form:

$$\log[\frac{p(y_i = 1|\boldsymbol{x}_i)}{1 - p(y_i = 1|\boldsymbol{x}_i)}] = \boldsymbol{w} \cdot \boldsymbol{x}_i + b, \tag{5.21}$$

where $\boldsymbol{w}$ is a vector of coefficients of corresponding $\boldsymbol{x}_i$, which represents the statistical significance level and b is a scalar. Eq. 5.21 can be transformed to:

$$p(y_i = 1|\boldsymbol{x}_i) = \frac{\exp^{(\boldsymbol{w} \cdot \boldsymbol{x}_i + b)}}{1 + \exp^{(\boldsymbol{w} \cdot \boldsymbol{x}_i + b)}}, \tag{5.22}$$

where $p(y_i = 1|\boldsymbol{x}_i)$ is the probability that $\boldsymbol{x}_i$ belongs to class 1. The parameters of logistic regression are estimated using maximum likelihood:

$$\max_{\boldsymbol{w}, b} \mathcal{L}(\boldsymbol{x}|\boldsymbol{w}, b) = \prod_{i=1}^{l} [y_i \cdot p(y_i = 1|\boldsymbol{x}_i) + (1 - y_i) \cdot (1 - p(y_i = 1|\boldsymbol{x}_i)]. \tag{5.23}$$

Once optimal parameters are obtained, the prediction model for a new $\boldsymbol{x}_*$ is given by:

$$p(y_* = 1|\boldsymbol{x}_*) = \frac{1}{1 + \exp^{-(\boldsymbol{w} \cdot \boldsymbol{x}_* + b)}}, \tag{5.24}$$

and $p(y_* = 0|\boldsymbol{x}_*) = 1 - p(y_* = 1|\boldsymbol{x}_*)$. By setting the cutoff value as 0.5, if $p(y_* = 1|\boldsymbol{x}_*) > 0.5$, the subject $\boldsymbol{x}_*$ is inferred as class 1. Otherwise, it is classified as class 0.

5.3.2 Dealing with class imbalance through classifier ensembles

Ensemble learning is a machine learning technique that trains a set of weak classifiers and combines their predictions to produce one optimal predictive model [83]. Traditional ensemble learning methods could be divided into two categories: parallel and sequential training. Bagging and Random Forest are classical parallel approaches [84]. They train classifiers in advance, then combine predictions of these weak classifiers by majority voting or weighted average. While sequential approaches improve the performance gradually by correcting misclassified instances previous classifiers have. In this work, parallel ensemble learning is adopted and the majority voting is used to generate the final result (Fig.5.4).

Traditional ensemble methods work efficiently when the input training data set is balanced. However, in many real-world problems, data sets are typically imbalanced, which has a serious impact on the performance of classifiers. Classifier algorithms that do not consider class imbalance tend to be overwhelmed by the majority class and ignore the minority class [85]. Here, the size of the training set is altered by randomly undersampling a smaller majority training set. Undersampling is adopted rather rather oversampling because we aim to obtain n diverse traning subsets. It makes the final prediction of our ensemble model robust after applying the majority voting. Also, ensemble classifiers play an important role in the feature selection algorithm introduced in section 5.3.3. On the other hand, ovesampling has some drawbacks, which make it not suitable for our prob-

lem. Normally, oversamping can be performed by increasing the amount minority class instances through repeating or producing synthetic data [86]. However, repeating existing data makes overfitting likely especially for small-size dataset [86]. Also, medical collaborators state that mixing synthetic data with real data together could destroy the cohort structure.

Then, the majority class is undersampled until it ends up with the same number of points as the minority class. n binary classification, there are two classes, $c \in \{+1, -1\}$. The training data set is denoted as:

$$X_{trn} = \bigcup_{c \in \{+1,-1\}} X_{trn,c}, \tag{5.25}$$

and $s_c = |X_{trn,c}|$ is the size of training data of class c. After doing random undersampling n times, the balanced training data sets $S^i_{trn}, i = 1, 2, ..., n$ are obtained by:

$$S^i_{trn} = S^i_{trn,+1} \cup S^i_{trn,-1}. \tag{5.26}$$

Suppose $c = -1$ is the minority class. The size of the minority class is denoted as s_{-1} and $S^i_{trn,-1} = X_{trn,-1}$, which means the full training set of the minority class is taken. Each random undersampling of the majority class has the size s_{-1}. The $S^i_{trn,+1}$ is generated by random sampling with replacement s_{-1} elements from $X_{trn,+1}$.

$$|S^i_{trn,-1}| = |S^i_{trn,+1}| = s_{-1}. \tag{5.27}$$

C^i is the classification model estimated using S^i_{trn}, $i = 1, ..., n$. Once a new data x_{test} is given, it will be tested by:

$$y^i(x_{test}) \in \{+1, -1\}, \tag{5.28}$$

and the final prediction is obtained via majority voting:

$$y(x_{test}) = Sign\left(\sum_{i=1}^{n} y^i(x_{test})\right). \tag{5.29}$$

All steps are listed in the pseudocode and illustrated in Fig 5.4 below:

- Training data set $X_{trn} = \bigcup_{c\in\{+1,-1\}} X_{trn,c}$, where $s_{+1} < s_{-1}$
- Randomly undersample n times from X_{trn} to obtain balanced training data sets $S^i_{trn}, i = 1, 2, ..., n$, where $|S^i_{trn,-1}| = |S^i_{trn,+1}| = s_{-1}$
- For i in n:
  - Train classifier C^i on S^i_{trn}
- For C^i in $\boldsymbol{C} = \{C^1, ..., C^n\}$:
  - A new data x_{test} is tested by $y^i(x_{test}) \in \{+1, -1\}$
- Final result is obtained via majority voting $y(x_{test}) = Sign\left(\sum_{i=1}^{n} y^i(x_{test})\right)$

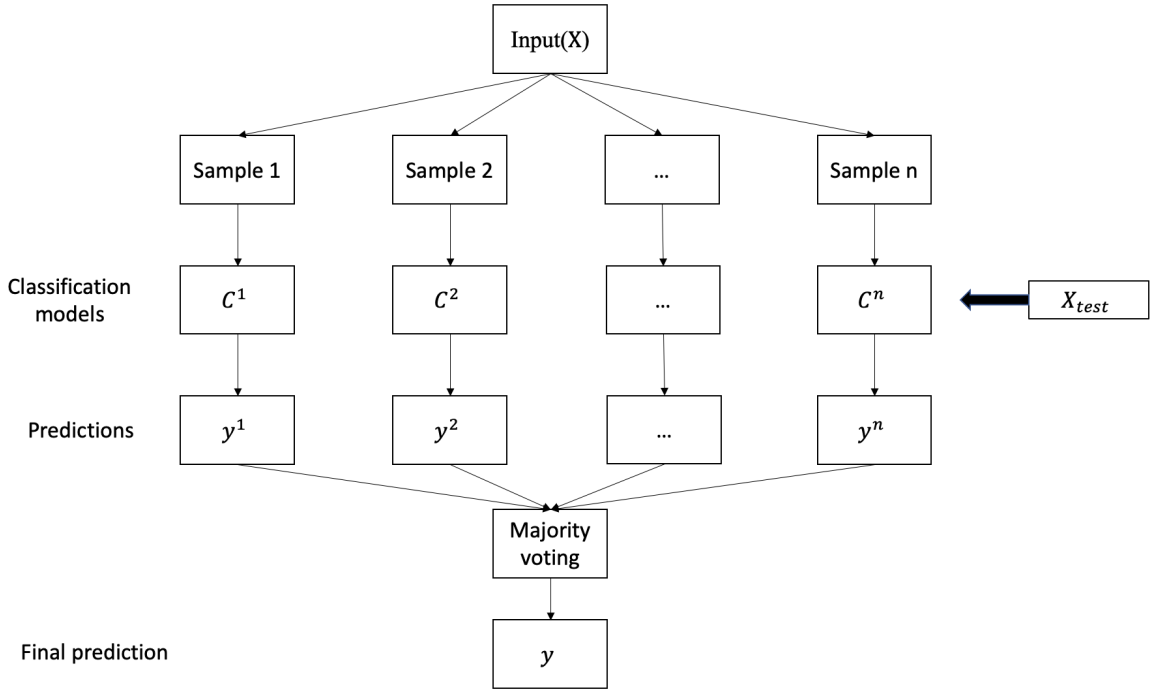


Figure 5.4: Ensemble and majority voting

5.3.3 The robust selection of important features

The importance plot is created based on the coefficients of SVM and logistic regression classifier models. In the signal space, features refer to time points. Whereas in the model space, features are model parameters, which are obtained through constrained MLE.

According to equations 5.18 and 5.21, $\boldsymbol{w}$ already corresponds to the importance value because of the function formation of $\boldsymbol{w}$ in classification models. Then, q significant classifiers with training accuracy over 85% from original n weak classifiers are picked. Considering the dimensionality of features is m, there is a set of importance values represented as:

$$W = \bigcup_{i,j} \{\boldsymbol{w}_j^i\} = \{w_1^1, ..., w_m^1, w_1^2, ..., w_m^2, ..., w_1^q, ..., w_m^q, \}, \tag{5.30}$$

where $i = 1, ..., q$ and $j = 1, ..., m$. Denote by Λ_p the p-percent quantile of values in W. For every feature $j = 1, 2, ..., m$, an indicator variable $Z_j^i = 1$ is introduced, if $w_j^i \geq \Lambda_p$, $Z_j^i = 0$ otherwise. The $p = 0.8$ is used. Then:

$$N_j = \sum_{i=1}^{q} Z_j^i, \tag{5.31}$$

which is the number of times feature j makes it into the set of important features ($\geq \Lambda_p$). The N_j is turned into normalised counts:

$$\widetilde{N}_j = \frac{N_j}{q}. \tag{5.32}$$

In the end, the feature is considered as important feature if $\widetilde{N}_j \geq \beta$, where $0 \leq \beta \leq 1$.

The β is nominally set to 0.4, although other values such as 0.6 are explored as well (see Appendix B).

5.4 Experimental methodology

According to Fig 5.5, all classification experiments are based on the three degrees of design freedom. In the first place, the classification is conducted in the signal space and model space respectively. Besides the full set of features, classifiers also work on the subset of features which is obtained by the robust selection of important features in section 5.3.3 in both learning spaces. In addition, classifiers are developed with both full and partial adrenal steroid pathway models.

5.4.1 Experiment design

Full set vs subset of features

In the signal space, features refer to measurements of metabolites at observational times. Imagining the time series $\boldsymbol{x}_i$, there are T_i features in total. The classification on the full set of features means that the classifier works on all T_i features. In order to reduce the data redundancy and explore discriminative features (observational times), a robust selection of important features is employed here (explained in section 5.3.3). Important features obtained by the feature selection are the subset of features. Then, classifiers are applied to that subset of features to compare with the results of the full set. The expectation is that the classification results of the subset would be similar to or at least not too much

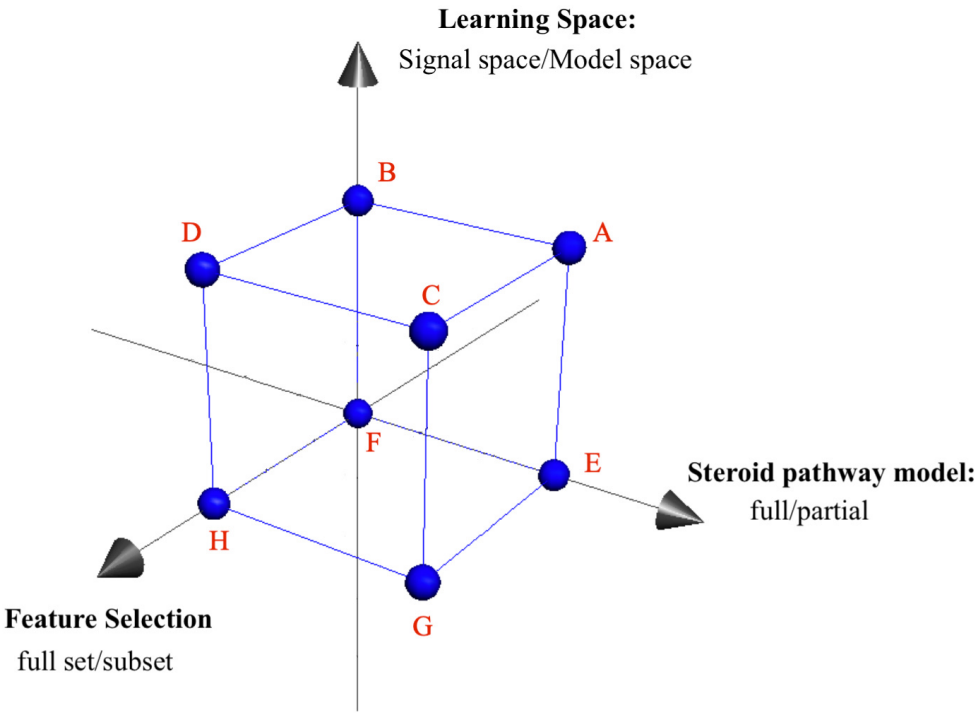


Figure 5.5: Three degrees of design freedom

worse than the results of the full set. The results of the subset would be even better than the full set if redundancy exists in the data. Information of the subset (important time points) is clinically meaningful because it helps to understand what time should be focused on. Hence, it can provide deep insights of certain conditions and assistance in diagnosis. It could also help to simplify the data sampling procedure by only obtaining samples in certain important time periods rather than sampling all 24 hours.

Features in the model space are model parameters of the inferential mechanistic model rather than values of metabolites in the signal space. The number of features is the dimensionality d of the parameter vector $\vec{\boldsymbol{\theta}}$ introduced in section 5.2.4. Classification

on the full set of features means that all d parameters are used in the classification. Just like the signal space, the feature selection based on importance plots in the model space is also conducted and then the classification with the subset of features (important parameters) is created. The expectation is that results of full set and subset features would be similar because biomedically only certain important time periods or model parameters can discriminate coresponding condition from control and it is not necessary to include all features. We also expect each disease condition (Cushing's or PrimAldo) could be dominated by their corresponding model parameters based on biomedical facts, which have been explained in Chapter 3.

Full vs partial steroid pathway model

In the signal space, classification using the full adrenal steroid pathway model means that all measurements of four metabolites Corticosterone(C), Aldosterone(A) Cortisol(F), and Cortisone(E) corresponding to Eqs. 5.6, 5.7, 5.8 and 5.9 are used. However, the partial model is different depending on different conditions. In the binary classification of Control versus Cushing's, the partial model refers to ODEs of Cortisol(F) and Cortisone(E) in Eqs. 5.8 and 5.9. Thus, only measurements of Cortisol and Cortisone from the Glucocorticoid pathway are used in the classification. In the classification of Control versus PrimAldo, the partial model is ODEs of Corticosterone(C), and Aldosterone(A) from the Mineralocorticoid pathway in Eqs. 5.6, 5.7. Hence, solely measurements of Corticosterone, Aldosterone are applied to the classifier.

In the model space, the classification with the full adrenal steroid pathway model means that all parameters in Eqs. 5.6, 5.7, 5.8 and 5.9 are used. The classification with partial

steroid pathway model for Control versus Cushing's employs model parameters in the Glucocorticoid pathway (Eqs. 5.8 and 5.9). For Control versus PrimAldo, model parameters in the Mineralocorticoid pathway (Eqs. 5.6 and 5.7) are used.

By doing classification using both full and partial models and comparing their classification performance, it will be clear if the partial pathway model is enough to separate disease conditions (Cushing's or PrimAldo) from Control or if the full pathway model is required.

5.4.2 Further experimental details

Before doing the classification, hyperparameter tuning for both classifier models SVM and logistic regression is conducted. Then, cross-validation is performed on the training set. All steps are listed in the pseudocode below.

- Randomly separate the data into training and testing set 100 times to have 100 splits
- Each i^{th} $(i = 1, \ldots, 100)$ contains training set X^i_{trn} and testing set X^i_{test}
- Define a set of hyper-parameter $\boldsymbol{C}$
- Randomly divide each X^i_{trn} in 100 training sets into training set for hyper-parameter tuning $X^i_{trn-hyper}$ and testing set $X^i_{test-hyper}$
- For parameter C in $\boldsymbol{C}$:
    - For $X^i_{trn-hyper}$ in 100 splits:
        - Generate balanced hyperparameter training set $S^{i,j}_{trn-hyper},\ j = 1, \ldots, 100$
        - Train model on $S^{i,j}_{trn-hyper}$
        - Evaluate model performance on $X^i_{test-hyper}$
    - Calculate average performance for parameter C
- Obtain the hyperparameter C_{best} that yields best average performance
- For X^i_{trn} in 100 splits:
    - Generate balanced training set $S^{i,j}_{trn}$ using method in section 5.3.2

- Train model on $S^{i,j}_{trn}$ with C_{best}
- Evaluate model performance on X^{i}_{test}

· Calculate average performance over 100 splits

In the first instance, the data set is split into training and testing set 100 times. Each testing set contains 5 Controls and 5 Conditions (Cushing's or PrimAldo). The remaining 95 Control, 17 Cushing's, and 13 PrimAldo are the training set $X^{i}_{trn}, i = 1, ..., 100$. For each X^{i}_{trn}, three data subjects of each class are separated as the testing set $X^{i}_{test-hyper}$. Then, the remaining of X^{i}_{trn} are undersampled to generate the balanced training set $S^{i,j}_{trn-hyper}, j = 1, ..., 100$ using the method in section 5.3.2 for hyperparameter tuning.

Once the best hyperparameter is obtained, the model is trained with C_{best} on $S^{i,j}_{trn}$. $S^{i,j}_{trn}$ is the balanced training set generated from X^{i}_{trn}. The training set X^{i}_{trn} is imbalanced with 95 Control, 17 Cushing's, and 13 PrimAldo. Then the repeated undersampling and classifier ensembles approach discussed in section 5.3.2 is applied to solve the class imbalance problem. In the end, each weak classifier has the training set with 34 data subjects (17 Control, 17 Cushing's) for Control versus Cushing's and 26 data subject (13 Control, 13 PrimAldo) for Control versus PrimAldo. The performance is evaluated on X^{i}_{test}.

Both SVM and logistic regression are trained in Python using scikit-learn. In SVM, the linear model is used and the hyperparameter C is set as 1 for Cushing's, and 100 for PrimAldo. In logistic regression, the solver 'liblinear' is adopted and hyperparameter C is set as 10 for Cushing's, and 100 for PrimAldo. Values of hyperparameter C are generated through the hyperparameter tuning explained above.

5.5 Results and discussion

All results presented in this section are based on the data set with 140 subjects (100 Control, 22 Cushing's, and 18 PrimAldo). The structure of this section follows the experiment design (three degrees of design freedom) in section 5.4.1. Results in both signal space and model space are demonstrated. Each space has classification models using full versus partial pathway model and in full set versus subset feature space. In the signal space, input data are measurements of four metabolises after the missing value imputation and starting time alignment. Input data in the model space are vectors of model parameters estimated from raw time series data (with missing values). Results obtained using model parameters estimated from imputed data are also shown in Appendix A. Confusion tables are used for classification performance analyses and evaluations. For the binary classification, the scheme of the confusion table is shown in Table 5.3. The robust selection of important features is illustrated using importance plots.

Table 5.3: Confusion table scheme

	True Control	True Cushing's
Predicted Control	True Negative (TN)	False Negative (FN)
Predicted Cushing's	False Positive (FP)	True Positive (TP)

5.5.1 Signal space

Signal space full steroid pathway model

Feature space - full set

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.88 +- 0.11	0.12 +- 0.11
Predicted Cushing's	0.07 +- 0.1	0.93 +- 0.1
Sensitivity	0.89	
Accuracy	0.91	

LR Cushing's	True Control	True Cushing's
Predicted Control	0.85 +- 0.14	0.15 +- 0.14
Predicted Cushing's	0.07 +- 0.12	0.93 +- 0.12
Sensitivity	0.86	
Accuracy	0.89	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.87 +- 0.14	0.13 +- 0.14
Predicted PrimAldo	0.17 +- 0.13	0.83 +- 0.13
Sensitivity	0.86	
Accuracy	0.85	

LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.88 +- 0.15	0.12 +- 0.15
Predicted PrimAldo	0.18 +- 0.18	0.82 +- 0.18
Sensitivity	0.87	
Accuracy	0.85	

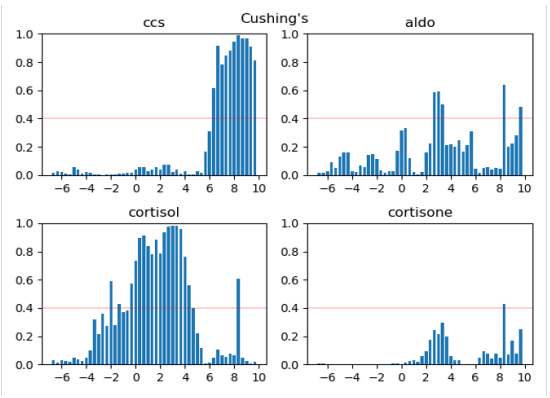
Feature space - subset

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.92 +- 0.09	0.08 +- 0.09
Predicted Cushing's	0.05 +- 0.09	0.95 +- 0.09
Sensitivity	0.92	
Accuracy	0.94	

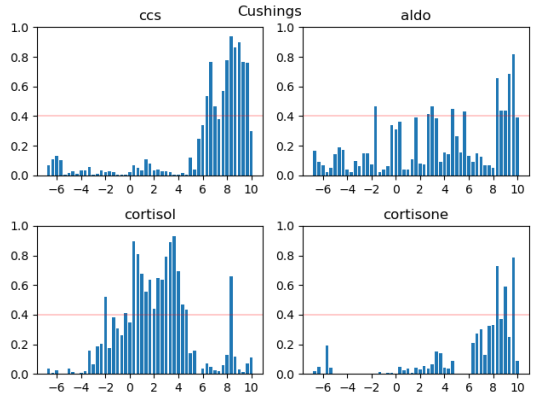
LR Cushing's	True Control	True Cushing's
Predicted Control	0.89 +- 0.11	0.11 +- 0.11
Predicted Cushing's	0.07 +- 0.1	0.93 +- 0.1
Sensitivity	0.89	
Accuracy	0.91	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.90 +- 0.12	0.10 +- 0.12
Predicted PrimAldo	0.17 +- 0.13	0.83 +- 0.13
Sensitivity	0.89	
Accuracy	0.87	

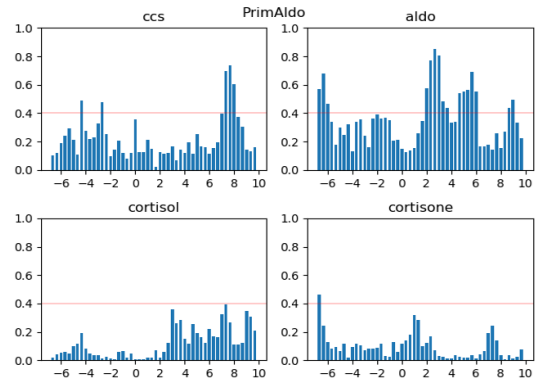
LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.83 +- 0.18	0.17 +- 0.18
Predicted PrimAldo	0.21 +- 0.17	0.79 +- 0.17
Sensitivity	0.82	
Accuracy	0.81	



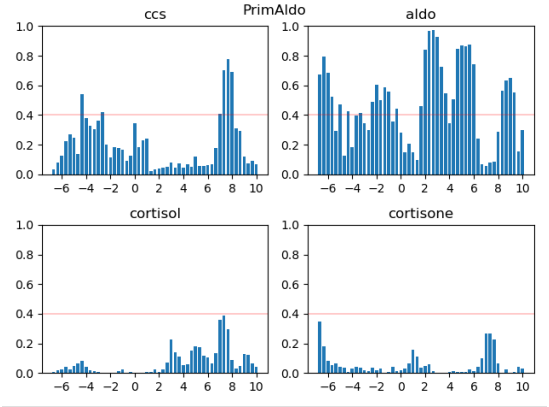
(a) svm signal Cushing's full



(b) lr signal Cushing's full



(c) svm signal PrimAldo full



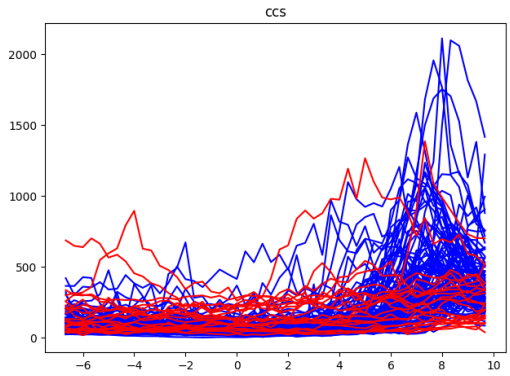
(d) lr signal PrimAldo full

Figure 5.6: Importance plots in the signal space with full adrenal steroid pathway model

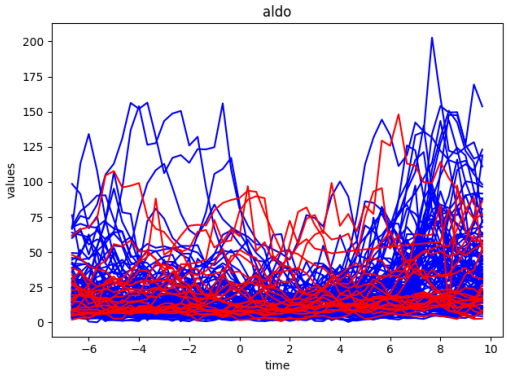
Confusion tables and importance plots in the signal space using the full adrenal steroid pathway model are illustrated above. Both full set and subset feature spaces have four confusion tables. Two of them are SVM and logistic regression for Control versus Cuhsing's and the other two are for Control versus PrimAldo. Classification in the full set feature space adopts all measurements from 5 pm to 10 am, while in the subset feature space, only selected important time points are included. Those important features are generated using the robust selection approach described in section 5.3.3 demonstrated on importance plots in Fig.5.6. The horizontal axis of each subplot in Fig. 5.6 refers to times. -7, 0, and 10 correspond to 5 pm, midnight, and 10 am respectively. The vertical axis from 0 to 1 corresponds to the importance values (normalised frequencies) of time points.

The threshold of 0.4 is picked and it means that features with normalised frequencies greater than 0.4 are considered as important features.

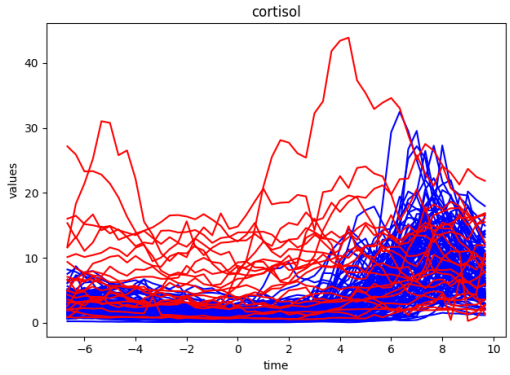
Generally, the classification performance of Control versus Cushing's is better than Control versus PrimAldo. This is the fact in medical settings that the trajectories of PrimAldo are much more similar to Control than Cushing's. Thus, it is understandable that the separation between Control and Cushing's is better than Control and PrimAldo. Further, classification results and importance plots of SVM and logistic regression are consistent in both full set and subset feature space.



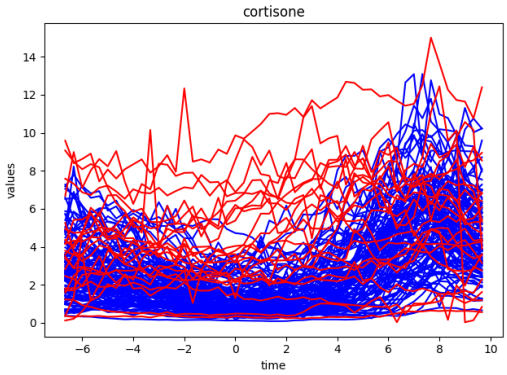
(a) Trajectories of CCS after imputation



(b) Trajectories of Aldo after imputation



(c) Trajectories of Cortisol after imputation



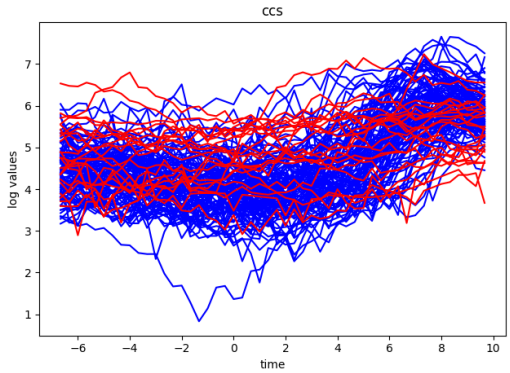
(d) Trajectories of Cortisone after imputation

Figure 5.7: Trajectories of four metabolises after imputation

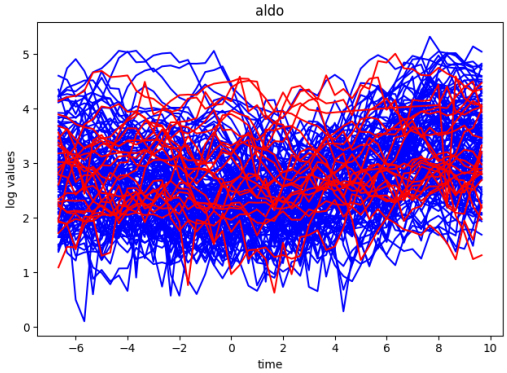
The SVM performance of Control versus PrimAldo in the full set and subset feature space is similar, while logistic regression is better in the full set feature space. It means that full features might be required to classify Control and PrimAldo. In addition, Fig. 5.6c and 5.6d show that important features of PrimAldo are mainly in Corticosterone (CCS) and Aldosterone (Aldo), which is exactly what the inferential biomedical model illustrates. PrimAldo is mainly caused by the excess of Aldo, which is correlated with CCS. Compared to the full feature space, the performance of Control versus Cushing's in the subset feature space is slightly better. Especially, accuracy rates of SVM in the subset feature space are very impressive with true negative 0.92 and true positive 0.95. This can be explained as some noisy features with importance values less than 0.4 are discarded. According to the importance plots in Fig. 5.6, important time points selected to separate Control and Cushing's are mainly CCS in the morning roughly from 7 am to 10 am, and Cortisol from 10 pm to 4 am (Fig.5.6a and 5.6b). It is surprising that Cushing's could be affected by CCS, which is in the mineralocorticoid pathway of the inferential biomedical model mainly constrains PirmAldo subjects. Hence, it is necessary to check the time series trajectories.

Trajectories shown in Fig.5.7 are time series of Control (blue) and Cushing's (red) after the univariate Gaussian process imputation. A log transformation is applied to measurements of all four metabolises to make them roughly under the same scale (Fig.5.8). The log transformation is used rather than standardisation rescale because it can keep the trajectories of the time series, which is the key for the classification in the signal space. And the boxplots of data after log transformation at each time point are listed in Fig.

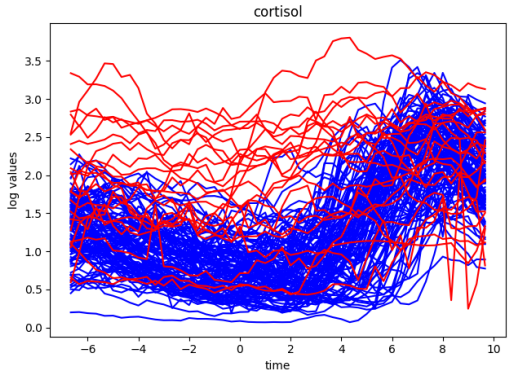
5.9. From those plots, it is clear that there is a good degree of separation of Control and Cushing's in Cortisol from about 8 pm to 4 am, which fits the finding of the importance plot of Cortisol. Cortisone is correlated to Cortisol so the situation is the same. It is not surprising to see that there is also a difference between Control and Cushing's in CCS during the early morning. Fig.5.9a shows that Control has a surge in CCS while Cushing's does not. This is also what the importance plot of CCS has exploded above. To sum up, apart from Cortisol and Cortisone, measurements of CCS in the early morning are also helpful to separate Control and Cushing's.



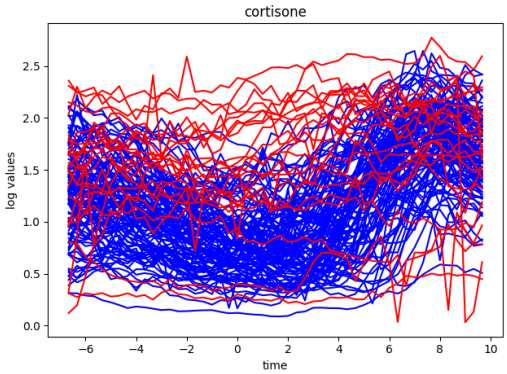
(a) Trajectories of CCS after imputation and log transformation



(b) Trajectories of Aldo after imputation and log transformation

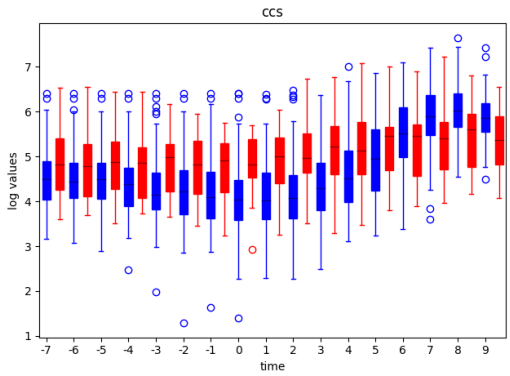


(c) Trajectories of Cortisol after imputation and log transformation

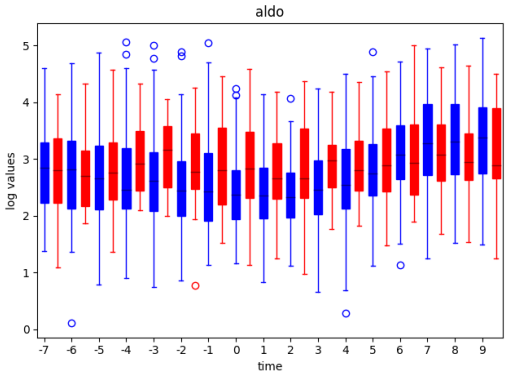


(d) Trajectories of Cortisone after imputation and log transformation

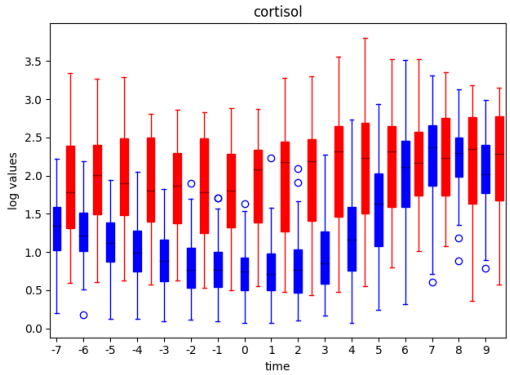
Figure 5.8: Trajectories of four metabolises after imputation and log transformation



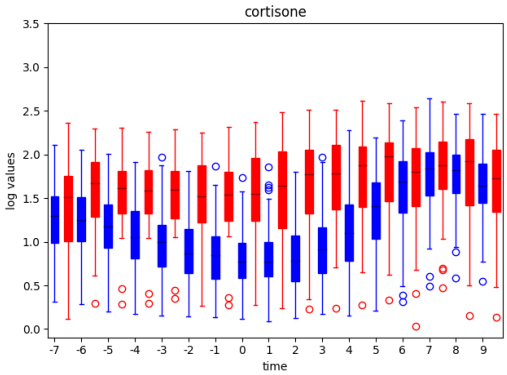
(a) Boxplot of CCS after imputation and log transformation



(b) Boxplot of Aldo after imputation and log transformation



(c) Boxplot of Cortisol after imputation and log transformation



(d) Boxplot of Cortisone after imputation and log transformation

Figure 5.9: Boxplot of four metabolises after imputation and log transformation

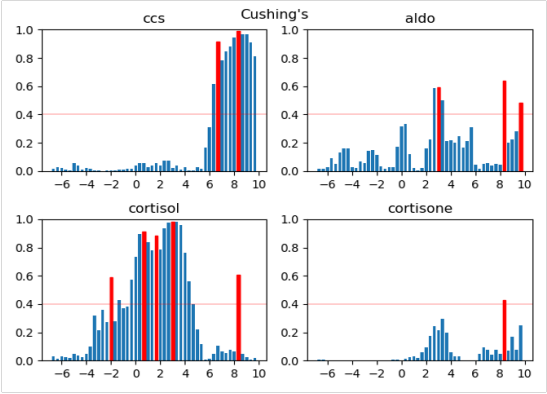
Feature space - peaks

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.91 +- 0.1	0.09 +- 0.1
Predicted Cushing's	0.05 +- 0.08	0.95 +- 0.08
Sensitivity	0.91	
Accuracy	0.93	

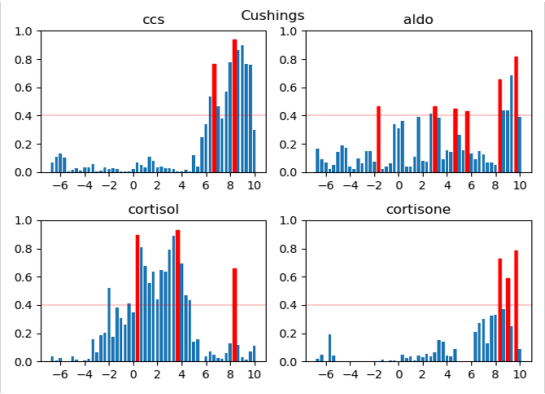
LR Cushing's	True Control	True Cushing's
Predicted Control	0.89 +- 0.11	0.11 +- 0.11
Predicted Cushing's	0.05 +- 0.1	0.95 +- 0.1
Sensitivity	0.90	
Accuracy	0.92	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.89 +- 0.12	0.11 +- 0.12
Predicted PrimAldo	0.17 +- 0.12	0.83 +- 0.12
Sensitivity	0.88	
Accuracy	0.86	

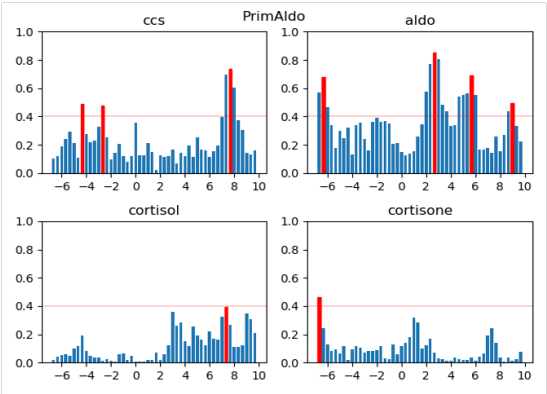
LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.83 +- 0.16	0.17 +- 0.16
Predicted PrimAldo	0.20 +- 0.18	0.80 +- 0.18
Sensitivity	0.82	
Accuracy	0.82	



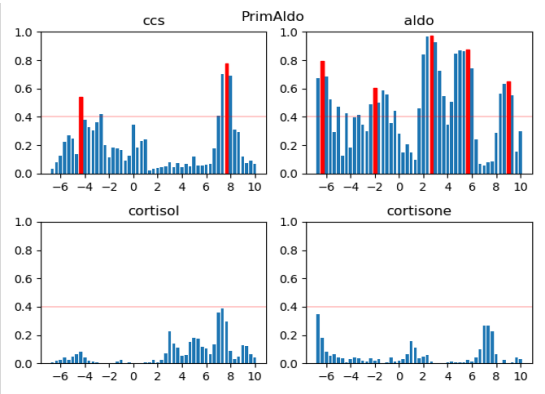
(a) svm Cushing's full on peaks



(b) lr Cushing's full on peaks



(c) svm PrimAldo full on peaks



(d) lr PrimAldo full on peaks

Figure 5.10: Importance plots in the signal space with partial steroid pathway model on selected peaks

Furthermore, peak features (highlighted in read in Fig. 5.10) are picked on the basis of the subset of features (features with normalised frequencies greater than 0.4). Due to the nature of biomedical data sampling, measurements around are correlated with each other so it is possible that an important peak feature could represent a chunk of time points around it. Therefore, it is worth training classifiers only with those few peak features. The classification performance of peak features shown in the confusion tables above is surprisingly good, especially for Cushing's with an accuracy rate of true positive 0.95. Consequently, peak features not only already have enough discriminative power, but also can provide information about what important time points should be focused on.

Signal space partial steroid pathway model

Feature space - full set

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.85 +- 0.11	0.15 +- 0.11
Predicted Cushing's	0.05 +- 0.09	0.95 +- 0.09
Sensitivity	0.86	
Accuracy	0.90	

LR Cushing's	True Control	True Cushing's
Predicted Control	0.77 +- 0.18	0.23 +- 0.18
Predicted Cushing's	0.06 +- 0.12	0.94 +- 0.12
Sensitivity	0.80	
Accuracy	0.86	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.91 +- 0.12	0.09 +- 0.12
Predicted PrimAldo	0.21 +- 0.13	0.79 +- 0.13
Sensitivity	0.90	
Accuracy	0.85	

LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.90 +- 0.12	0.10 +- 0.12
Predicted PrimAldo	0.22 +- 0.17	0.78 +- 0.17
Sensitivity	0.89	
Accuracy	0.84	

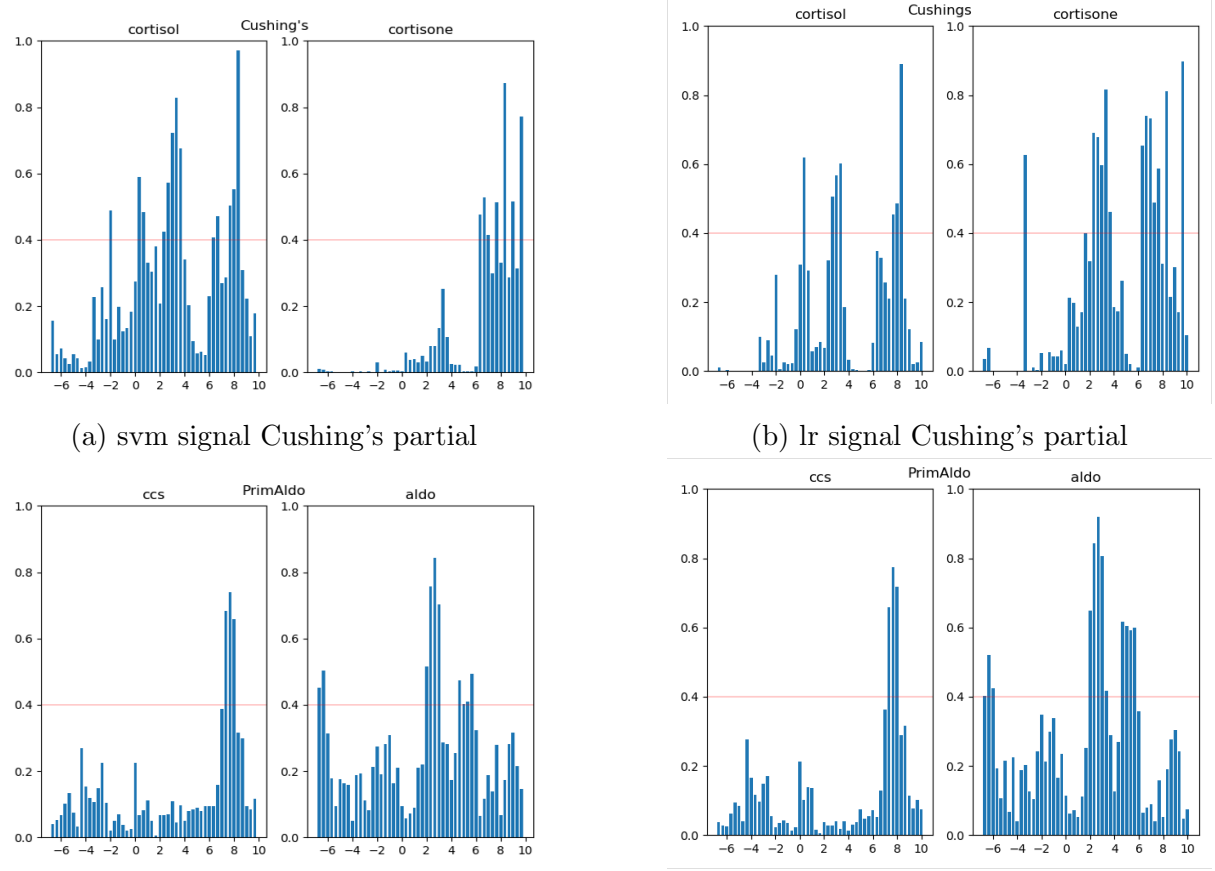
Feature space - subset

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.87 +- 0.11	0.13 +- 0.11
Predicted Cushing's	0.05 +- 0.08	0.95 +- 0.08
Sensitivity	0.88	
Accuracy	0.91	

LR Cushing's	True Control	True Cushing's
Predicted Control	0.84 +- 0.16	0.16 +- 0.16
Predicted Cushing's	0.06 +- 0.11	0.94 +- 0.11
Sensitivity	0.85	
Accuracy	0.89	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.85 +- 0.11	0.15 +- 0.11
Predicted PrimAldo	0.16 +- 0.13	0.84 +- 0.13
Sensitivity	0.85	
Accuracy	0.85	

LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.82 +- 0.16	0.18 +- 0.16
Predicted PrimAldo	0.21 +- 0.17	0.79 +- 0.17
Sensitivity	0.81	
Accuracy	0.81	



(a) svm signal Cushing's partial

(b) lr signal Cushing's partial

(c) svm signal PrimAldo partial

(d) lr signal PrimAldo partial

Figure 5.11: Importance plots in the signal space with partial adrenal steroid pathway model

This section discusses the classification performance in the signal space using a partial adrenal steroid pathway model (left model with Cortisol and Cortisone for Cushing's; right model with CCS and Aldo for PrimAldo).

In the case of Cushing's, accuracy rates of true positive are 0.95 and 0.94, which are as good as accuracy rates in the full pathway model section, while rates of true negative are slightly worse. This can be explained as the classification here on Control versus Cushing's is constrained by the left pathway (with no CCS), while CCS is helpful to separate Control and Cushing's (discussed in the previous section). Instead of CCS, features of

Cortisol and Cortisone roughly from 6 am to 9 am/10 am are picked as important features in Fig. 5.11a and 5.11b, which are not selected in the full pathway model setting. However, the performance is not as good as using CCS in the early morning because it is clear that trajectory plots show measurements of CCS during the early morning can separate Control and Cushing's better than measurements of Cortisol or Cortisone from the same time period. PrimAldo is mainly dominated by CCS and Aldo (right pathway) so the performance of PrimAldo using the partial pathway model is very similar to the performance using the full pathway model. However, the accuracy rates in the subset feature space are slightly worse than in the full set feature space. It confirms that full features are needed to classify Control and PrimAldo.

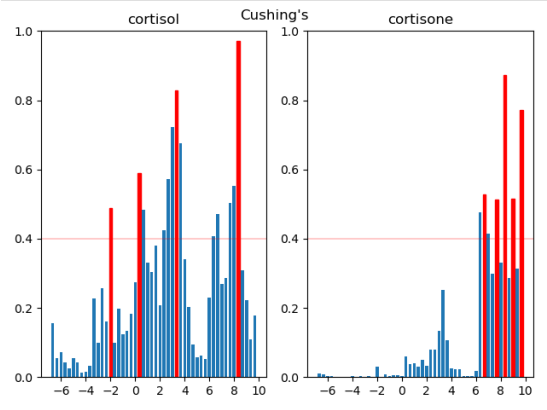
Feature space - peaks

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.83 +- 0.13	0.17 +- 0.13
Predicted Cushing's	0.03 +- 0.12	0.97 +- 0.12
Sensitivity	0.85	
Accuracy	0.90	

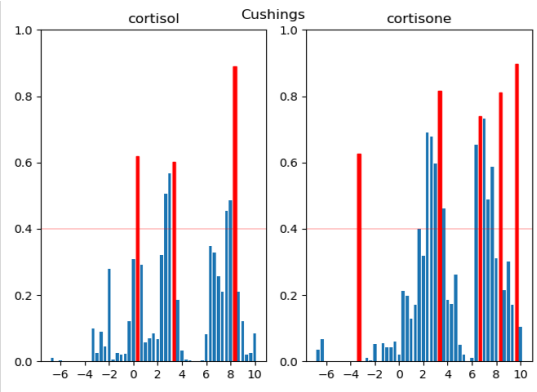
LR Cushing's	True Control	True Cushing's
Predicted Control	0.84 +- 0.14	0.16 +- 0.14
Predicted Cushing's	0.05 +- 0.08	0.95 +- 0.08
Sensitivity	0.86	
Accuracy	0.90	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.86 +- 0.13	0.14 +- 0.13
Predicted PrimAldo	0.17 +- 0.14	0.83 +- 0.14
Sensitivity	0.86	
Accuracy	0.85	

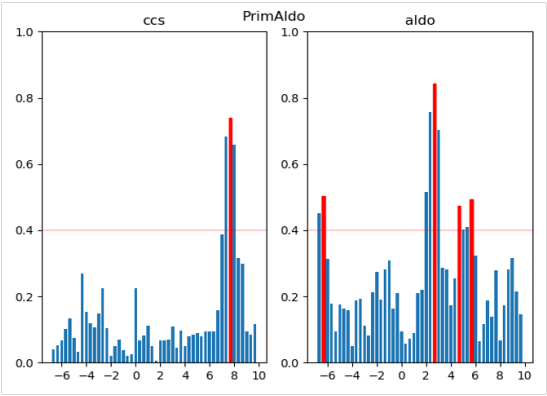
LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.85 +- 0.13	0.15 +- 0.13
Predicted PrimAldo	0.21 +- 0.17	0.79 +- 0.17
Sensitivity	0.84	
Accuracy	0.82	



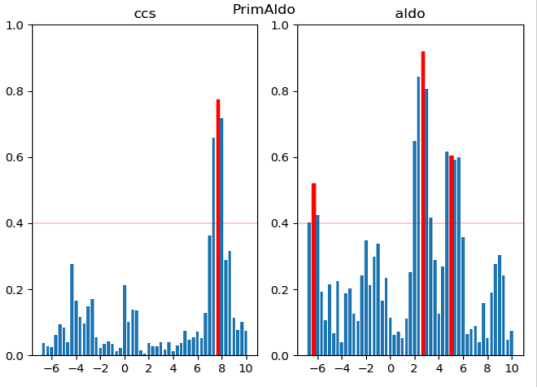
(a) svm Cushing's partial on peaks



(b) lr Cushing's partial on peaks



(c) svm PrimAldo partial on peaks



(d) lr PrimAldo partial on peaks

Figure 5.12: Importance plots in the signal space with partial steroid pathway model on selected peaks

Results on peak features using the partial pathway model are even more surprising with very high accuracy rates of true positive (SVM 0.97, logistic Regression 0.95) for Cushing's. Also, the classification performance of PrimAldo is impressive because it is trained only on four time points see Fig. 5.12c and 5.12d.

After comparing and evaluating all confusion tables and importance plots in the signal space, key findings are listed below:

· In general, the performance of SVM and logistic regression is consistent, not only classification results showed in confusion tables but also feature selection results de-

rived from the importance plots. These two classification models based on different principles share similar performance, which illustrates that the results are robust and reliable.

- Binary classifiers work better on Control versus Cushing's than Control versus PrimAldo. This is confirmed by medical collaborators because PrimAldo is more similar with Control in trajectories than Cushing's.

- CCS is crucial to separate Control and Cushing's so the partial pathway model (left model without CCS) is not enough to classify Control and Cushing's. This is a surprising finding because biomedically Cushing's is mainly caused by the excess of Cortisol and Cortisone, while CCS is related to PrimAldo. However, results reveal that CCS can also affect Cushing's especially during the early morning.

- Classification of Control versus Cushing's in the subset feature space performs better than the full feature space, which means there might be redundancy in the data.

- Classification of Control versus PrimAldo requires full features because it is harder to separate these two with similar trajectories.

- Classification using peak features (only a few time points) can still have reasonable and comparable results. This is meaningful for future sampling. Medical people can only measure some important time points rather than 24 hours.

5.5.2 Model space

Model space full steroid pathway model

Feature space - full set

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.83 +- 0.12	0.17 +- 0.12
Predicted Cushing's	0.08 +- 0.11	0.92 +- 0.11
Sensitivity	0.84	
Accuracy	0.88	

LR Cushing's	True Control	True Cushing's
Predicted Control	0.80 +- 0.13	0.20 +- 0.13
Predicted Cushing's	0.10 +- 0.13	0.90 +- 0.13
Sensitivity	0.82	
Accuracy	0.85	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.79 +- 0.14	0.21 +- 0.14
Predicted PrimAldo	0.19 +- 0.16	0.81 +- 0.16
Sensitivity	0.77	
Accuracy	0.80	

LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.78 +- 0.15	0.22 +- 0.15
Predicted PrimAldo	0.19 +- 0.15	0.81 +- 0.15
Sensitivity	0.79	
Accuracy	0.80	

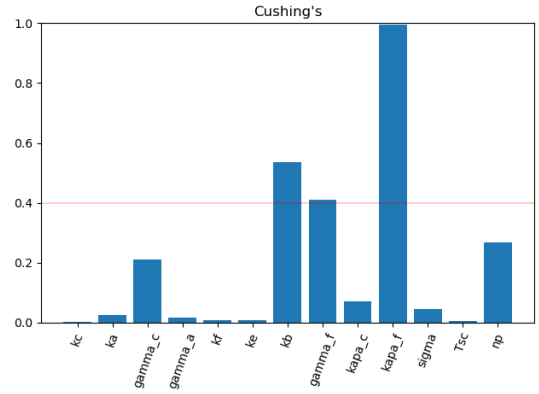
Feature space - subset

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.81 +- 0.12	0.19 +- 0.12
Predicted Cushing's	0.01 +- 0.05	0.99 +- 0.05
Sensitivity	0.84	
Accuracy	0.90	

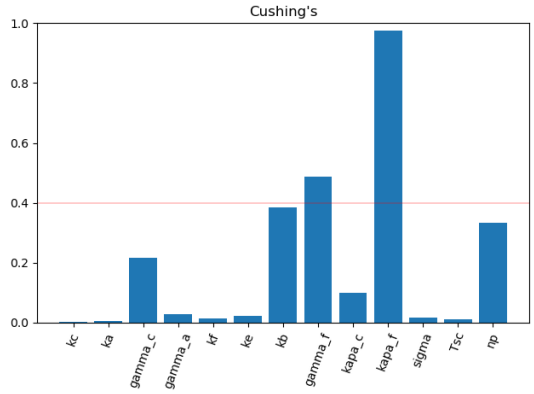
LR Cushing's	True Control	True Cushing's
Predicted Control	0.80 +- 0.12	0.20 +- 0.12
Predicted Cushing's	0.02 +- 0.06	0.98 +- 0.06
Sensitivity	0.83	
Accuracy	0.89	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.75 +- 0.16	0.25 +- 0.16
Predicted PrimAldo	0.23 +- 0.15	0.77 +- 0.15
Sensitivity	0.75	
Accuracy	0.76	

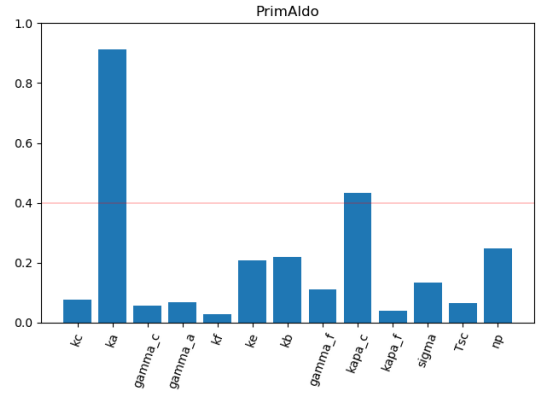
LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.70 +- 0.13	0.30 +- 0.13
Predicted PrimAldo	0.22 +- 0.18	0.78 +- 0.18
Sensitivity	0.72	
Accuracy	0.74	



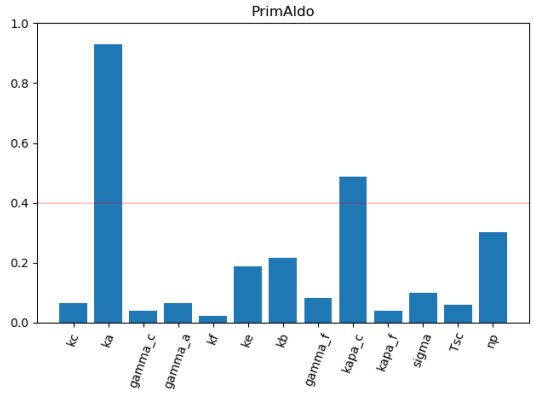
(a) svm model Cushing's full



(b) lr model Cushing's full



(c) svm model PrimAldo full



(d) lr model PrimAldo full

Figure 5.13: Importance plots in the model space with full steroid pathway model

In the model space, input data are model parameter vectors of the inferential mechanistic model, which has thirteen free parameters. Features in the model space refer to these model parameters. The full feature space of full adrenal steroid models contains all thirteen parameters. Cushing's subset feature space of full adrenal steroid models in SVM involves three parameters K_b, $Gamma_f$, and $Kappa_f$ (Fig. 5.13a), while the subset feature space of of full adrenal steroid models in logistic regression has two parameters $Gamma_f$ and $Kappa_f$ (Fig. 5.13b). The subset feature space of PrimAldo includes two parameters K_a and $Kappa_c$ (both SVM and logistic regression). The subset feature space is generated by the feature selection. Parameters with importance values greater than 0.4 are considered as important parameters (features) and compose the subset feature

space. Compared to the signal space, the performance in the model space of Control versus Cushing's in the full feature space is slightly worse both in true positive and true negative. However, true positive accuracy rates (Cushing's) in the subset feature space are very impressive, which are 0.99 in SVM and 0.98 in logistic regression. PrimAldo has better performance in the full feature space rather than the subset feature space. It means that the subset feature space with only K_a and $Kappa_c$ is not enough to classify Control and PrimAldo.

Model space partial steroid pathway model

Feature space - full set

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.82 +- 0.11	0.18 +- 0.11
Predicted Cushing's	0.02 +- 0.06	0.98 +- 0.06
Sensitivity	0.84	
Accuracy	0.90	

LR Cushing's	True Control	True Cushing's
Predicted Control	0.80 +- 0.11	0.20 +- 0.11
Predicted Cushing's	0.02 +- 0.05	0.98 +- 0.05
Sensitivity	0.83	
Accuracy	0.89	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.80 +- 0.15	0.20 +- 0.15
Predicted PrimAldo	0.19 +- 0.16	0.81 +- 0.16
Sensitivity	0.80	
Accuracy	0.81	

LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.81 +- 0.15	0.19 +- 0.15
Predicted PrimAldo	0.20 +- 0.15	0.80 +- 0.15
Sensitivity	0.81	
Accuracy	0.81	

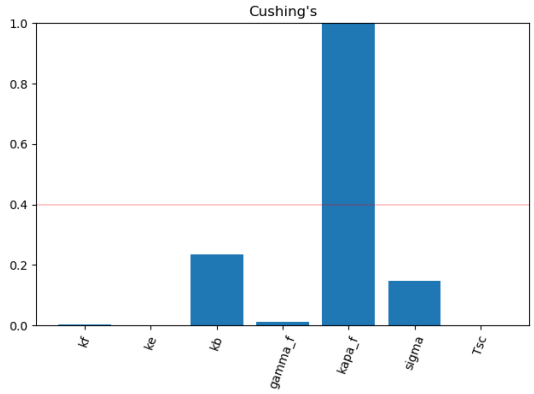
Feature space - subset

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.80 +- 0.11	0.20 +- 0.11
Predicted Cushing's	0.02 +- 0.07	0.98 +- 0.07
Sensitivity	0.83	
Accuracy	0.89	

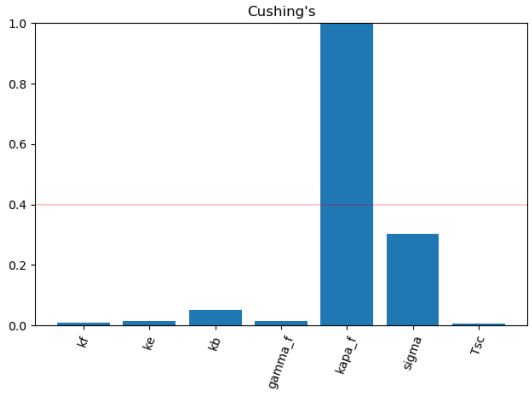
LR Cushing's	True Control	True Cushing's
Predicted Control	0.82 +- 0.12	0.18 +- 0.12
Predicted Cushing's	0.02 +- 0.05	0.98 +- 0.05
Sensitivity	0.84	
Accuracy	0.90	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.70 +- 0.11	0.30 +- 0.11
Predicted PrimAldo	0.23 +- 0.18	0.77 +- 0.18
Sensitivity	0.72	
Accuracy	0.74	

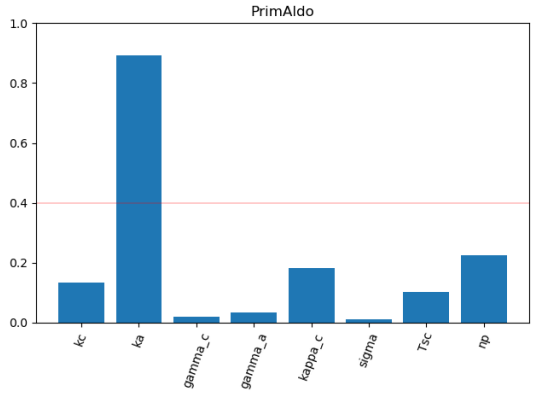
LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.69 +- 0.14	0.31 +- 0.14
Predicted PrimAldo	0.25 +- 0.18	0.75 +- 0.18
Sensitivity	0.71	
Accuracy	0.72	



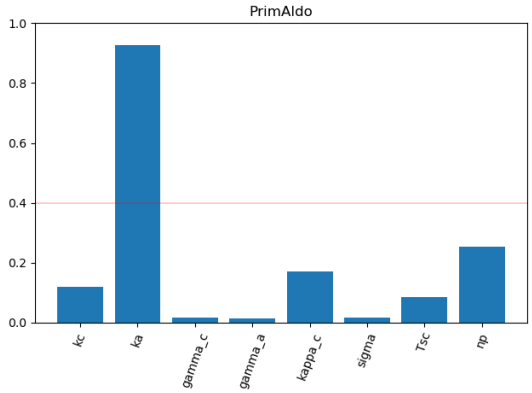
(a) svm model Cushing's partial



(b) lr model Cushing's partial



(c) svm model PrimAldo partial



(d) lr model PrimAldo partial

Figure 5.14: Importance plots in the model space with partial steroid pathway model

The partial adrenal steroid pathway model of Cushing's refers to the left model containing only Cortisol and Cortisone. The partial model of PrimAldo is the right model including CCS and Aldo. There are seven model parameters in the left model and eight parameters in the right model. The greatest discovery in this section with the partial adrenal steroid pathway model is the performance in the subset feature space of Cushing's. The accuracy rate of 0.98 of true positive is very impressive and referring to Fig.5.14a and Fig.5.14b, this result is based on just one parameter $Kappa_f$, although the accuracy rate of true negative is just around 0.8. Then, a hybrid model is proposed for Cushing's to improve the true negative accuracy. The hybrid classifier combines the classification model on $Kappa_f$ in the model space and another model in the signal space, which has to have a

high accuracy rate in the true negative. Details and confusion tables of the hybrid model are shown in the next section.

$Kappa_f$ is the coefficient of the convex combination of circadian drive in Cortisol. Generally, Control should have a higher $Kappa_f$ value than Cushing's because it has an oscillation in Cortisol during the early morning. Thus, $Kappa_f$ is the most important parameter to separate Control and Cushing's. However, Cushing's is complex with several subgroups and some of them could not be detected using $Kappa_f$ only.

Hybrid model

The hybrid model (only for Cushing's) is a combination of two base classifiers. Those two base classifiers picked have to have good classification performance and have to be complementary. After comparing a number of classifiers, the first classifier adopted is from the model space, which is only trained on $Kappa_f$. It is discussed above with a true positive accuracy rate of 0.98 in both SVM and logistic regression. Another classifier picked is from the signal space using the full adrenal steroid pathway model in the subset feature space because it has comparatively high true negative accuracy rates (SVM 0.92; logistic Regression 0.89). Once a test subject comes to the hybrid model, if the model space confirms it is a Cushing's then it is labeled as Cushing's. Otherwise, we take the result of the signal space.

Although the true positive accuracy rate in Table 5.4 slightly decreases to 0.97 (SVM) and 0.94 (logistic regression) respectively, the true negative accuracy increases to 0.98 (SVM) and 0.97 (logistic regression), which is much more meaningful in the medical point of

view. A higher true negative rate associated with a lower false positive rate means less misdiagnosis of Cushing's patients.

Table 5.4: Hybrid model combining the model space with the signal space

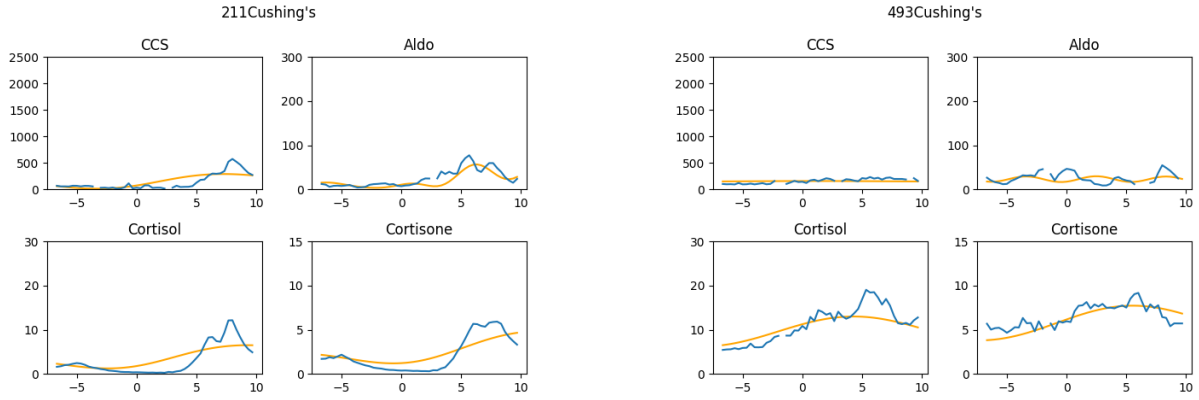
SVM Cushing's	True Control	True Cushing's
Predicted Control	0.98 +- 0.06	0.02 +- 0.06
Predicted Cushing's	0.03 +- 0.06	0.97 +- 0.06
Sensitivity	0.98	
Accuracy	0.98	

LR Cushing's	True Control	True Cushing's
Predicted Control	0.97 +- 0.07	0.03 +- 0.07
Predicted Cushing's	0.06 +- 0.1	0.94 +- 0.1
Sensitivity	0.97	
Accuracy	0.96	

Table 5.5: Testing results of new Cushing's in SVM

SID	Model space $Kappa_f$	Signal space full model subspace	Hybrid
211	0	1	1
486	1	1	1
493	0	1	1
579	1	0	1
605	1	1	1
626	1	1	1
722	1	1	1
726	1	1	1

However, the testing accuracy of the hybrid model is based on test subjects used by two base classifiers. Therefore, it is necessary to use another 8 additional Cushing's subjects for verification. Those additional Cushing's subjects do not appear in previous classifiers' training and testing. Testing results of SVM (results of logistic regression are similar) are shown in Table 5.5 (0 refers to Control and 1 refers to Cushing's). They are tested using the hybrid model with a 100% accuracy rate. The classification model only with $Kappa_f$ wrongly classifies 211 and 493 as Control. Those two Cushing's subjects have large values of $Kappa_f$ indeed (surge in the early morning in Cortisol in Fig.5.15a and 5.15b) but they might belong to a subtype of Cushing's, whose rhythm is very similar to Control. Thus, only using $Kappa_f$ is not adequate to separate those Cushing's subjects from Controls.



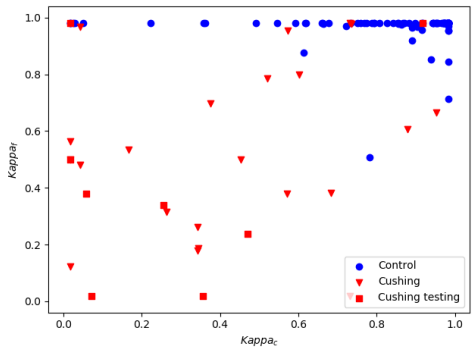
(a) Cushing's 211

(b) Cushing's 493

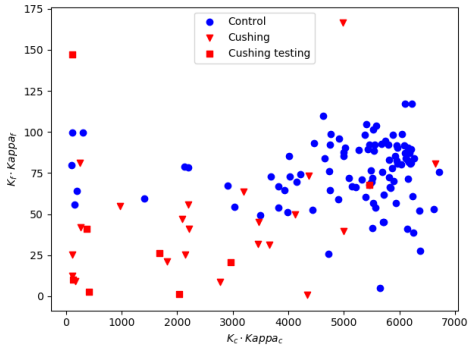
Figure 5.15: Plots of wrongly predicted subjects. Blues lines in 5.15a, 5.15b are trajectories of original data. Original lines are trajectories solved by parameter estimations

Point estimations in the model space

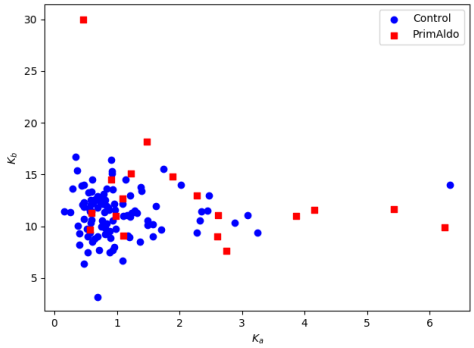
Point estimates of parameters are directly related to the classification performance in the model space. The point estimation map (Fig. 5.16a) and box plot (Fig. 5.17c) of $Kappa_f$ reveal that $Kappa_f$ is significant to distinguish between Control and Cushing's subjects. According to the box plot, most Control subjects have larger values of $Kappa_f$ than Cushing's. There are also some Cushing's subjects that have high $Kappa_f$ values. They might belong to a subtype of Cushing's. The K_c and K_f are Corticosterone and Cortisol synthesis rates respectively. $K_c \times Kappa_c$ and $K_f \times Kappa_f$ control the effects of circadian drives on CCS and Cortisol after taking synthesis rates into account. Values of them control the overall amplitudes of oscillations. Normally Control subjects have high values of $K_c \times Kappa_c$ and $K_f \times Kappa_f$ than Cushing's (see Fig. 5.16b). However, the degree of separation in the point estimation maps of PrimAldo is not as good as Cushing's, which agrees with the confusion tables discussed above.



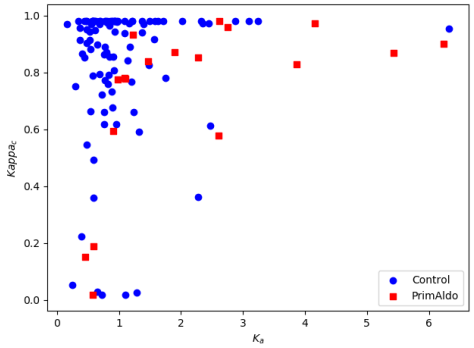
(a) $kappa_c$ and $kappa_f$



(b) $k_c \cdot kappa_c$ and $k_f \cdot kappa_f$

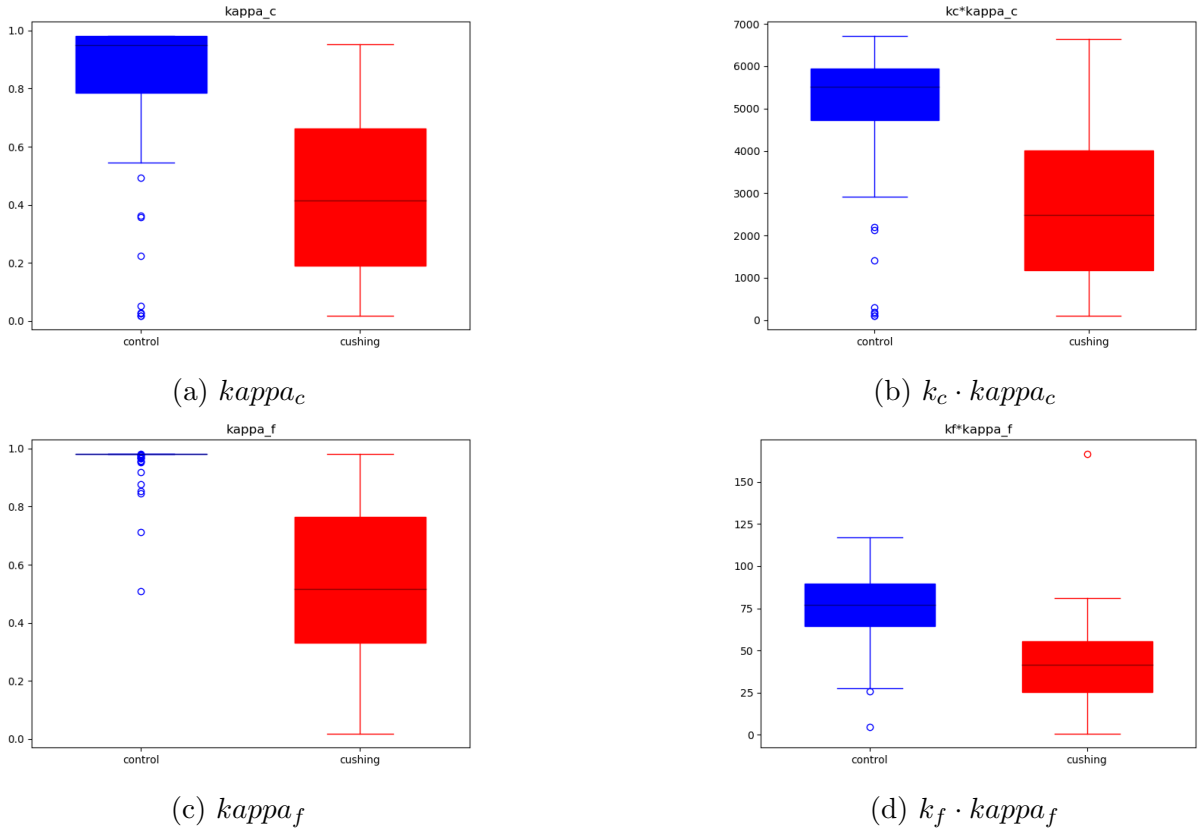


(c) k_a and k_b



(d) k_a and $kappa_c$

Figure 5.16: Maps of point estimations of parameters



(a) $kappa_c$

(b) $k_c \cdot kappa_c$

(c) $kappa_f$

(d) $k_f \cdot kappa_f$

Figure 5.17: Box plots of important parameters

Key findings of classification results in the model space and the comparison between signal space and model space classification are listed below:

- $Kappa_f$ the coefficient of the convex combination of circadian drive in Cortisol is an important parameter to separate Control and Cushing's.
- A hybrid classifier is developed for Control versus Cushing's to improve the overall classification performance by combining classifiers in the model space and signal space.
- All parameters are required to classify Control and PrimAldo because it is much harder to separate PrimAldo from Control than Cushing's.

· Generally, the classification performance in the model space is similar with the signal space in Control versus Cushing's and the model space is a little worse than signal space. However, learning in the model space directly works on raw data without missing value imputation, which is a clear advantage than learning in the signal space. Also, learning in the model space includes domain knowledge in the classification model, which is an expected requirement from medical experts.

5.6 Unfiltered data set contains large observational gaps

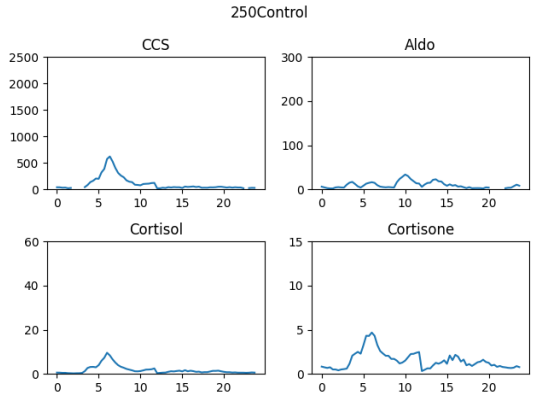
5.6.1 Unfiltered data set information

In previous experiments, the real data set used contains 140 subjects with 100 Control, 22 Cushing's, and 18 PrimAldo. It is a filtered data set containing data subjects with only a few missing values such as subjects in Fig 5.18, which the uni-variate Gaussian process can work well on. Our medical collaborators also provide some data subjects containing large observation gaps for example subjects with one or even two metabolises missing (see Fig 5.19 and 5.20). From the biomedical point of view, it is more desirable to include those subjects containing substantial missing values. Thus, the unfiltered data set with 270 subjects (217 Controls, 31 Cushing's, 22 PrimAldo) is used. It includes all data that have been collected except five subjects with both Corticosterone and Aldosterone missing (Fig 5.20). Corticosterone and Aldosterone are correlated to each other so it would be even harder to estimate their values or model parameters if both of them are missing. After discussing with medical collaborators, those five subjects are discarded.

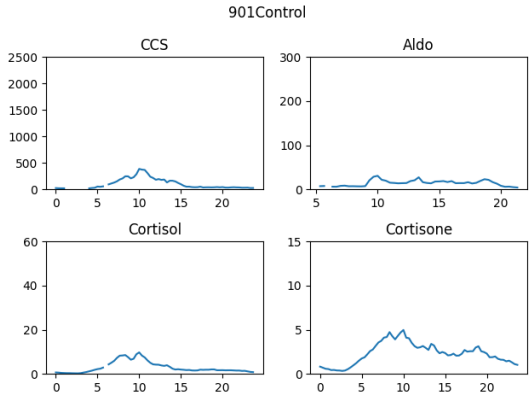
In biomedical terms, a missing data rate of 5% or less is inconsequential [87]. Thus, it is acceptable to remove those five subjects because 5 out of 275 is much less than 5%.

However, the univariate Gaussian process is not able to impute the whole missing metabolite. It means that getting the accurate imputation for the whole missing metabolise is very difficult so doing classification directly in the signal space with the unfiltered data set is hard. Fortunately, this can be done in the model space. Also, initial values of four metabolites are treated as free parameters rather than assigned by first measurements of all four metabolites because time series with whole metabolite missing has no measurements of the missing metabolite. Also, it is more accurate and reasonable to treat them as free parameters. Then, the adjoint method is employed to compute the gradient of the log-likelihood constrained by the initial-value ODEs. The adjont method is much more efficient than sensitivity equation approach, which we used for filtered dataset.

In the unfiltered data set, the periodic boundary condition is applied to align the data rather than removing measurements of certain time periods. The periodic boundary condition is typical where something is repeated many times but the optimisation or simulation only needs to take place over one cycle of that sequence. An example of a repeating process is the body's natural circadian rhythm. Therefore, time series are shifted to ensure the starting time of each time series is 0 (midnight).

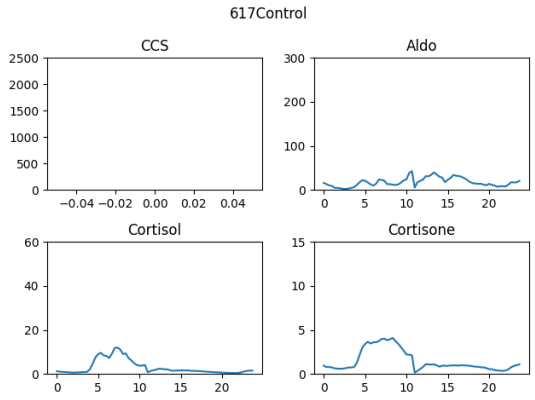


(a) SID 250 Control

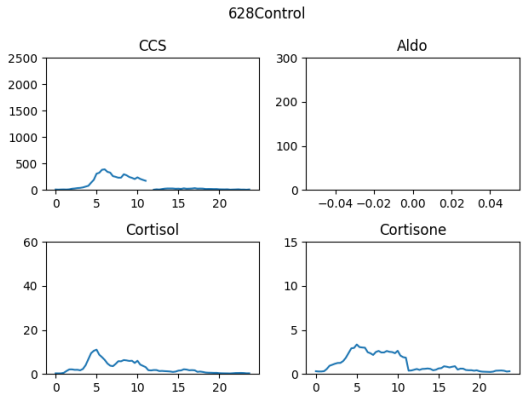


(b) SID 901 Control

Figure 5.18: Subjects with few observations missing

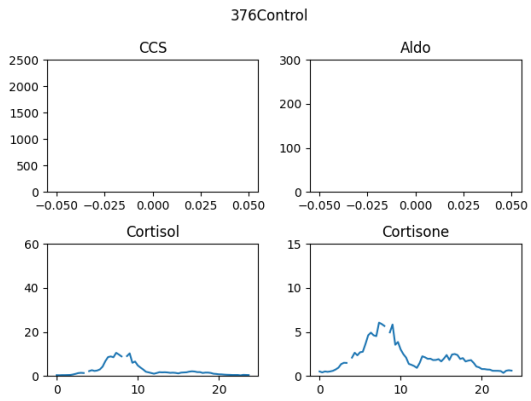


(a) SID 617 Control

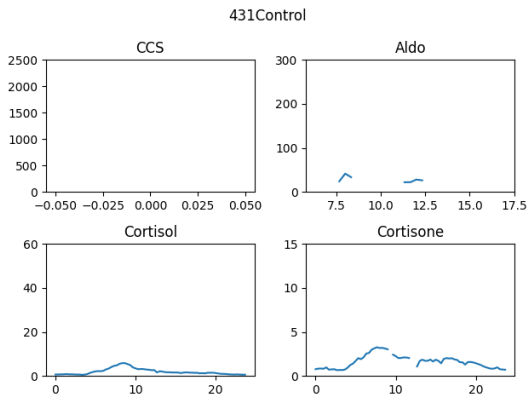


(b) SID 628 Control

Figure 5.19: Subjects with one metabolise missing



(a) SID 376 Control



(b) SID 431 Control

Figure 5.20: Subjects with more than one metabolises missing

5.6.2 The adjoint method

The adjoint method is useful to obtain the gradient of a function constrained by an initial-value ODE. In our case, the mechanistic model is ODEs with unknown initial conditions and 13 other unknown parameters. The adjoint method is employed here to obtain and optimize the initial values of four metabolises together with the other 13 free parameters, which are required to run the forward model (inferential mechanistic model) with the ODE solver.

Consider the problem:

$$\min_{p} \int_0^T f(x,p)\ dt \equiv F(x,p), \tag{5.33}$$

$$\textbf{\textit{s.t.}} \quad h(x,\dot{x},p,t) = 0, \tag{5.34}$$

$$x(0) = x_0(p), \tag{5.35}$$

where p is a vector of the free parameter of the underlying model. x is the state variable and it is a function of time. $h(x,\dot{x},p,t) = 0$ is an ODE in implicit form. $x(0)$ is the initial condition and $x(0) = x_0(p)$ shows that $x(0)$ is a function of the free parameters p. For the data fitting application, the objective can be:

$$\frac{1}{2}\int_0^T (x(t) - x_{data}(t))^{\mathsf{T}}(x(t) - x_{data}(t))\ dt, \tag{5.36}$$

where x_{data} is obtained via solving the forward model (ODE) using the ODE solver.

A gradient-based optimization algorithm requires the calculation of the total deriva-

tive/gradient:

$$d_p F(x,p) = \int_0^T \partial_x f \, d_p x + \partial_p f \; dt, \tag{5.37}$$

where ∂_x and ∂_p denote a partial derivative with respect to x and p respectively and d_p is a total derivative. It is difficult to calculate $d_p x$ unless x is available in closed form. If x is not, there are two common approaches aiming to evade having to calculate it. The first approach is simply to approximate the gradient $d_p F(x,p)$ by finite differences over p. The second approach that the adjoint method uses, is to develop a second ODE [88].

The first step of the adjoint method is to develop the Lagrangian corresponding to the optimization problem:

$$\mathcal{L} \equiv \int_0^T f(x,p) - \boldsymbol{\lambda}^\mathsf{T} h(x,\dot{x},p,t) dt - \mu^\mathsf{T}(x(0) - x_0(p)). \tag{5.38}$$

Because $h(x,\dot{x},p,t) = 0$ and $x(0) - x_0(p) = 0$, $d_p\mathcal{L} \equiv d_p F$. The total derivative is:

$$d_p\mathcal{L} = \int_0^T \partial_x f \, d_p x + \partial_p f - \boldsymbol{\lambda}^\mathsf{T}(\partial_x h d_p x + \partial_{\dot{x}} h d_p \dot{x} + \partial_p h) \; dt - \mu^\mathsf{T}(1-1). \tag{5.39}$$

As the derivative of the term containing μ is zero, it is not used. The integrand in the total derivative contains terms $d_p x$ and $d_p \dot{x}$. Then, the next step is to integrate by parts to get rid of the second one:

$$\int_0^T \boldsymbol{\lambda}^\mathsf{T} \partial_{\dot{x}} h d_p \dot{x}\, dt = \int_0^T \boldsymbol{\lambda}^\mathsf{T} d_t(\partial_{\dot{x}} h d_p x)\, dt \tag{5.40}$$

$$= \boldsymbol{\lambda}^\mathsf{T} \partial_{\dot{x}} h d_p x|_0^T - \int_0^T \dot{\boldsymbol{\lambda}}^\mathsf{T} \partial_{\dot{x}} h d_p x\, dt. \tag{5.41}$$

Substitute the result into Eq. 5.39 and rearrange it by collecting terms in $d_p x$:

$$d_p \mathcal{L} = \int_0^T (\partial_x f - \boldsymbol{\lambda}^\mathsf{T} \partial_x h + \dot{\boldsymbol{\lambda}}^\mathsf{T} \partial_{\dot{x}} h) d_p x + f_p - \boldsymbol{\lambda}^\mathsf{T} \partial_p h\, dt - \boldsymbol{\lambda}^\mathsf{T} \partial_{\dot{x}} h d_p x|_0^T. \tag{5.42}$$

Then set $\boldsymbol{\lambda}(T) = 0$ to make the entire term zero because $d_p x(T)$ is difficult to calculate. The $d_p x(0)$ is easy to calculate because $x_0(p)$ is known and it is just $\partial_p x_0(p)$. Also, set $\partial_x f - \boldsymbol{\lambda}^\mathsf{T} \partial_x h + \dot{\boldsymbol{\lambda}}^\mathsf{T} \partial_{\dot{x}} h = 0$ to avoid $d_p x$.

The algorithm is stated as follows:

- Integrate $h(x, \dot{x}, p, t) = 0$ for x_0 to x_T with initial condition $x(0) = x_0(p)$.
- Integrate $\partial_x f - \boldsymbol{\lambda}^\mathsf{T} \partial_x h + \dot{\boldsymbol{\lambda}}^\mathsf{T} \partial_{\dot{x}} h = 0$ for $\boldsymbol{\lambda}_T$ to $\boldsymbol{\lambda}_0$ with initial condition $\boldsymbol{\lambda}(T) = 0$.
- $d_p F = \int_0^T f_p - \boldsymbol{\lambda}^\mathsf{T} \partial_p h\, dt + \boldsymbol{\lambda}^\mathsf{T} \partial_{\dot{x}} h d_p x|_0$.

5.7 Results and discussion of unfiltered data set

Classification results of both SVM and logistic regression models in the model space are also presented on the unfiltered data set following the experiment design discussed in section 5.4.1. Classifiers are trained using both full and partial adrenal steroid pathway models. Also, important features obtained by the robust feature selection approach are demonstrated via importance plots.

5.7.1 Model space

Model space full steroid pathway model

Feature space - full set

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.90 +- 0.12	0.10 +- 0.12
Predicted Cushing's	0.06 +- 0.12	0.94 +- 0.12
Sensitivity	0.90	
Accuracy	0.92	

LR Cushing's	True Control	True Cushing's
Predicted Control	0.88 +- 0.12	0.22 +- 0.12
Predicted Cushing's	0.07 +- 0.11	0.93 +- 0.11
Sensitivity	0.81	
Accuracy	0.91	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.80 +- 0.14	0.20 +- 0.14
Predicted PrimAldo	0.19 +- 0.14	0.81 +- 0.14
Sensitivity	0.80	
Accuracy	0.81	

LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.77 +- 0.16	0.23 +- 0.16
Predicted PrimAldo	0.20 +- 0.15	0.80 +- 0.15
Sensitivity	0.78	
Accuracy	0.79	

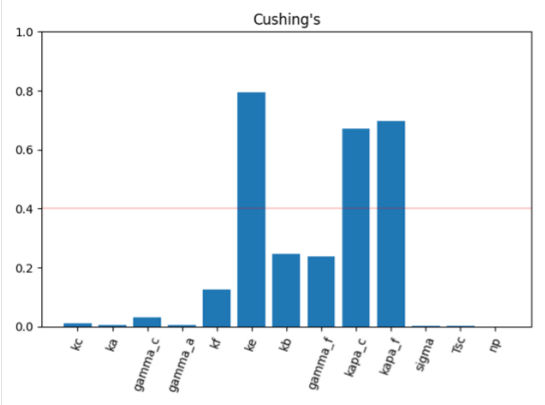
Feature space - subset

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.88 +- 0.12	0.12 +- 0.12
Predicted Cushing's	0.09 +- 0.11	0.91 +- 0.11
Sensitivity	0.88	
Accuracy	0.90	

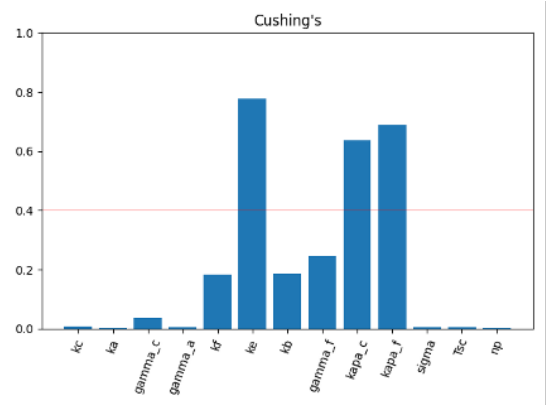
LR Cushing's	True Control	True Cushing's
Predicted Control	0.87 +- 0.11	0.13 +- 0.11
Predicted Cushing's	0.06 +- 0.09	0.94 +- 0.09
Sensitivity	0.88	
Accuracy	0.91	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.75 +- 0.16	0.25 +- 0.16
Predicted PrimAldo	0.23 +- 0.15	0.77 +- 0.15
Sensitivity	0.75	
Accuracy	0.76	

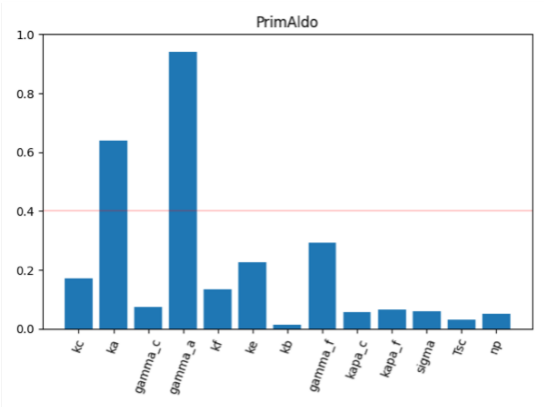
LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.72 +- 0.16	0.28 +- 0.16
Predicted PrimAldo	0.22 +- 0.14	0.78 +- 0.14
Sensitivity	0.74	
Accuracy	0.75	



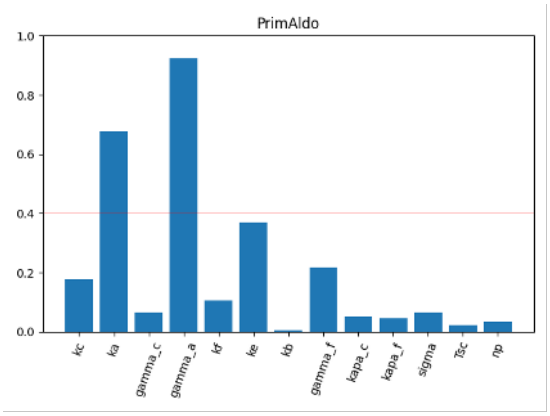
(a) svm model Cushing's full



(b) lr model Cushing's full



(c) svm model PrimAldo full



(d) lr model PrimAldo full

Figure 5.21: Importance plots in the model space with full adrenal steroid pathway model

This section is about the classification performance in the model space using the full adrenal steroid pathway model. The full set feature space here refers to all thirteen model parameters and the subset feature space is made up of important features with frequency values greater than 0.4 in Fig. 5.21. Then, it shows that the subset feature space of Cushing's has three important parameters ($K_e, Kappa_c$ and $Kappa_f$) and PrimAldo has 2 parameters (K_a and $Gamma_a$) in both SVM and logistic regression.

The performance in the subset feature space is similar to the full feature space, which is remarkable because it is only based on three or two parameters, while the full space contains thirteen parameters. In addition, the performance of SVM and logistic regression

is still consistent not just in confusion tables but also in importance plots (Fig. 5.21), from which both of them picked the same important features for both Cushing's and PrimAldo.

Model space partial steroid pathway model

Feature space - full set

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.87 +- 0.12	0.13 +- 0.12
Predicted Cushing's	0.04 +- 0.10	0.92 +- 0.10
Sensitivity	0.88	
Accuracy	0.90	

LR Cushing's	True Control	True Cushing's
Predicted Control	0.89 +- 0.12	0.11 +- 0.12
Predicted Cushing's	0.10 +- 0.11	0.90 +- 0.11
Sensitivity	0.89	
Accuracy	0.90	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.72 +- 0.13	0.28 +- 0.13
Predicted PrimAldo	0.28 +- 0.12	0.72 +- 0.12
Sensitivity	0.72	
Accuracy	0.72	

LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.70 +- 0.15	0.30 +- 0.15
Predicted PrimAldo	0.29 +- 0.12	0.71 +- 0.12
Sensitivity	0.70	
Accuracy	0.71	

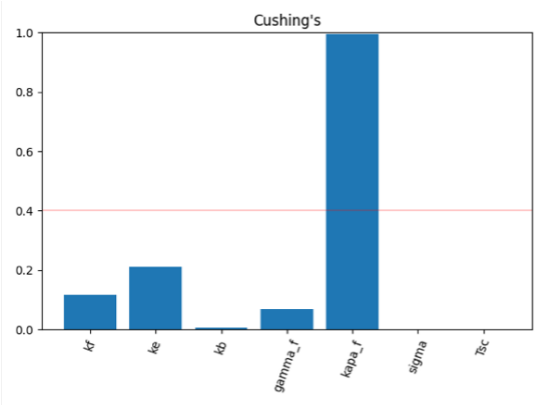
Feature space - subset

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.80 +- 0.10	0.20 +- 0.10
Predicted Cushing's	0.04 +- 0.10	0.96 +- 0.10
Sensitivity	0.83	
Accuracy	0.88	

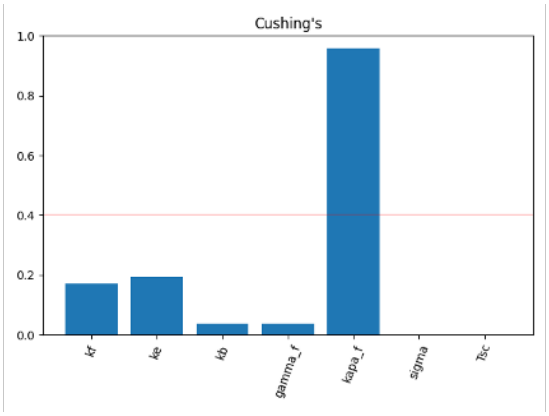
LR Cushing's	True Control	True Cushing's
Predicted Control	0.78 +- 0.12	0.22 +- 0.12
Predicted Cushing's	0.07 +- 0.12	0.93 +- 0.12
Sensitivity	0.81	
Accuracy	0.86	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.75 +- 0.16	0.25 +- 0.16
Predicted PrimAldo	0.23 +- 0.15	0.77 +- 0.15
Sensitivity	0.75	
Accuracy	0.76	

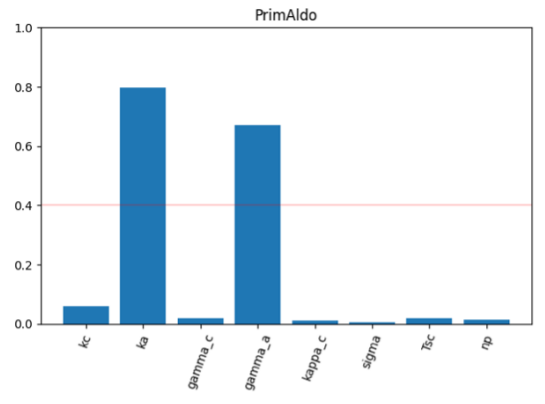
LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.72 +- 0.16	0.28 +- 0.16
Predicted PrimAldo	0.22 +- 0.14	0.78 +- 0.14
Sensitivity	0.74	
Accuracy	0.75	



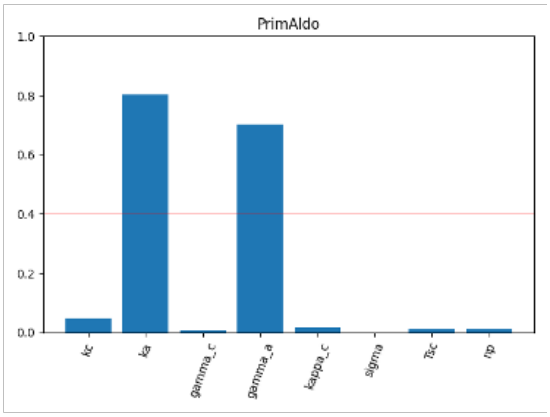
(a) svm model crushing's partial



(b) lr model cushing's partial



(c) svm model primaldo partial



(d) lr model primaldo partial

Figure 5.22: Importance plots in the model space with partial adrenal steroid pathway model

In general, the results of the partial adrenal steroid pathway model are not as good as the full model for both Cushing's and PrimAldo. However, it is worth pointing out that the true positive accuracy rates of Cushing's in the subset feature space are quite high (SVM 0.96 and logistic regression 0.93). Also, it is surprising such high accuracy rates are just based on one parameter $Kappa_f$. Therefore, a hybrid model is created by combining the classifier on $Kappa_f$ and another classifier on all thirteen parameters, which has a high accuracy rate of true negative (details in next section). The results of the feature selection of PrimAldo in full and partial pathway models are the same. Both have 2 important parameters which are K_e and $Gamma_a$. Thus, confusion tables are also the same.

Hybrid model of unfiltered data set

There is no classifier model developed in the signal space for the unfiltered data set because of large observational gaps. Hence, the hybrid model developed here is a combination of two classifiers both in the model space. They are classifiers on $Kappa_f$ and on all thirteen parameters respectively. The classifier on $Kappa_f$ has impressive accuracy rates of true positive (SVM 0.96 and logistic regression 0.93). The classifier on all thirteen parameters has true negative accuracy rates of 0.91 (SVM) and 0.88 (logistic regression). The confusion tables of hybrid classifiers of both SVM and logistic regression in Table 5.6 show that the accuracy rates of true negative are improved and this is the initial intention of developing the hybrid model.

Table 5.6: Hybrid model combining the model space with signal space

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.91 +- 0.07	0.09 +- 0.07
Predicted Cushing's	0.05 +- 0.11	0.95 +- 0.11
Sensitivity	0.91	
Accuracy	0.93	

LR Cushing's	True Control	True Cushing's
Predicted Control	0.90 +- 0.09	0.10 +- 0.09
Predicted Cushing's	0.09 +- 0.10	0.91 +- 0.10
Sensitivity	0.90	
Accuracy	0.91	

Moreover, there are eight Cushing's subjects not included in the training phase that are used for case studies to evaluate the performance of the hybrid model. SVM results of case studies can be seen in Table 5.7 (results of logistic regression are similar). The hybrid classifier, together with classifiers on $Kappa_f$ and thirteen parameters all misclassify Cushing's 211 to Control. Medical collaborators confirm that Cushing's 211 is an outlier that has a high $Kappa_f$ value and a similar trajectory as Control. The classifier of $Kappa_f$ also labels Cushing's 722 as Control (in Fig.5.23). It can also be explained because there is a surge in Cortisol around early morning, which corresponds to a large value of $Kappa_f$. However, the figure of Cushing's 722 illustrates that there is a gap between measurements around 3 pm (15 in the horizontal axis). This is the drawback of the periodic condition, which may also cause misclassification.

Table 5.7: Unfiltered dataset: Testing results of new Cushing's in SVM

SID	Model space $Kappa_f$	Model space 13 parameters	Hybrid
211	0	0	0
486	1	1	1
493	1	1	1
579	1	1	1
605	1	1	1
626	1	1	1
722	0	1	1
726	1	1	1

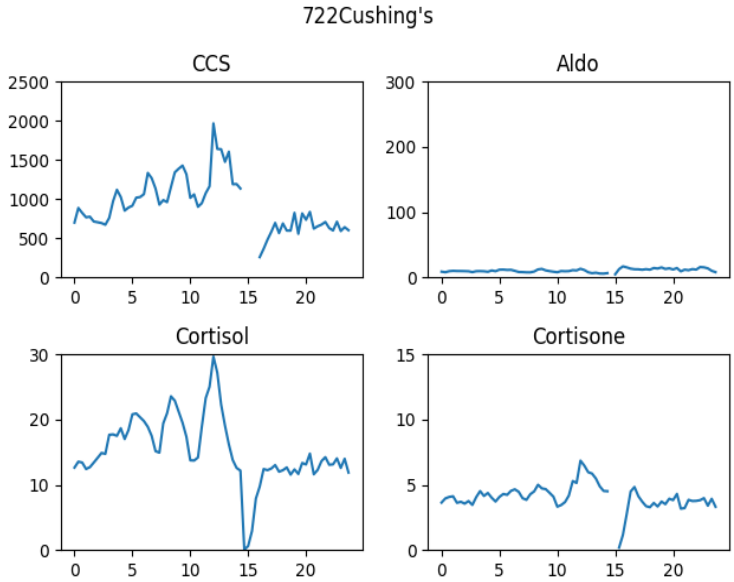
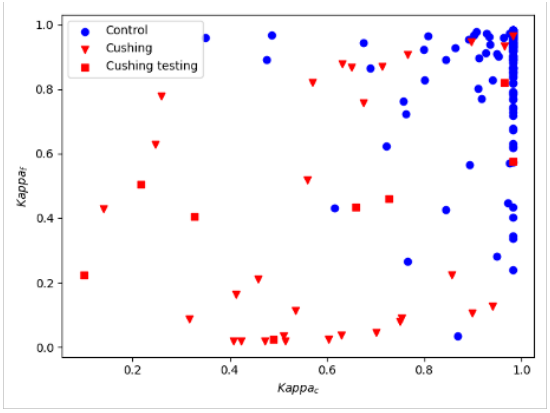
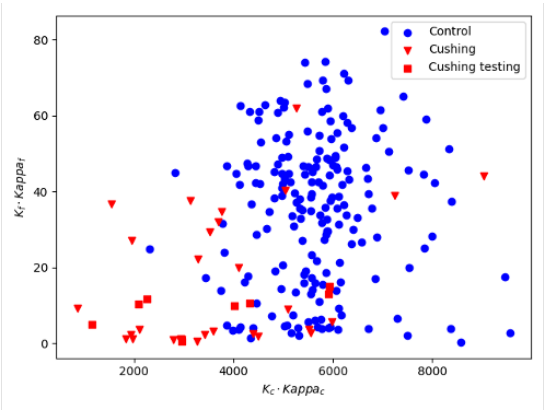


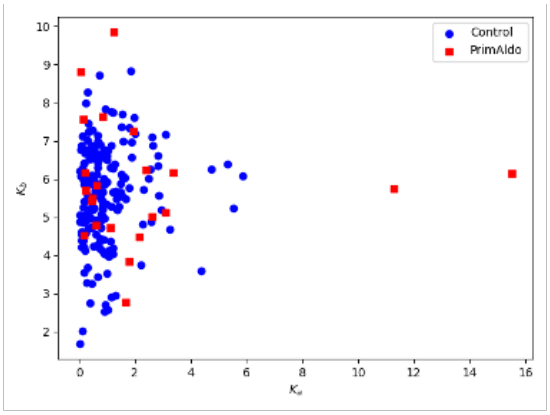
Figure 5.23: SID 722 Cushing's



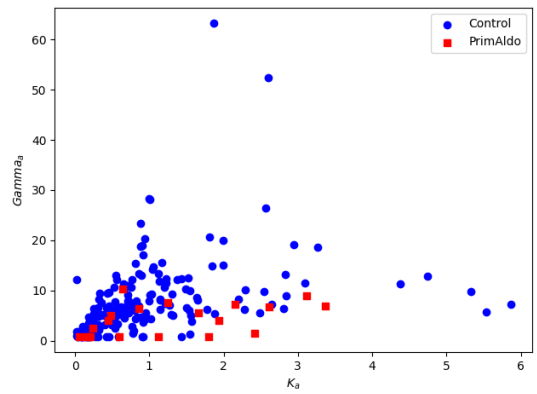
(a) $kappa_c$ and $kappa_f$



(b) $k_c \cdot kappa_c$ and $k_f \cdot kappa_f$



(c) k_a and k_b



(d) k_a and $gamma_a$

Figure 5.24: Maps of point estimations of parameters

Point estimations in the model space

This section illustrates maps of point estimations and box plots of some consequential parameters, which can help to interpret classification results by taking the mechanistic information into account.

Subplots 5.24a and 5.24b are maps corresponding to Cushing's subjects. The eight Cushing's for testing are marked as the red squares in plots 5.24a and 5.24b. Both plots reveal a good degree of separation of Cushing's from Control subjects. It is consistent with the feature selection of Cushing's, which shows parameters $Kappa_c$ and $Kappa_f$ are important. There are some Cushing's subjects mixed with Controls on the right side of both plots. Trajectories of two of them with SID 211 and 235 are plotted in Fig. 5.25. Blue lines are original trajectories and orange lines are trajectories generated using point estimations of parameters. It is obvious that there are oscillations in the morning in both CCS and Cortisol (211 and 235), which look more like Control than Cushing's. Clinical experts also verified that Control 211 is the patient with a diagnosis of adrenal Cushing's (a subtype of Cushing's). Their rhythms are very similar to Control and the abnormalities are subtle. Then, it is not surprising that they are mixed with Controls.

In addition, the degree of separation would also be good if only $Kappa_c$ is used to train the classifier according to Fig.5.24a. Confusion tables are in Table 5.8. The accuracy rates of true positive of both SVM and logistic regression are as good as the classifiers based on $Kappa_f$. The box plots in Fig. 5.26 draw the same conclusion. Blue boxes and red boxes refer to Control and Cushing's respectively. Clearly, Control subjects have high

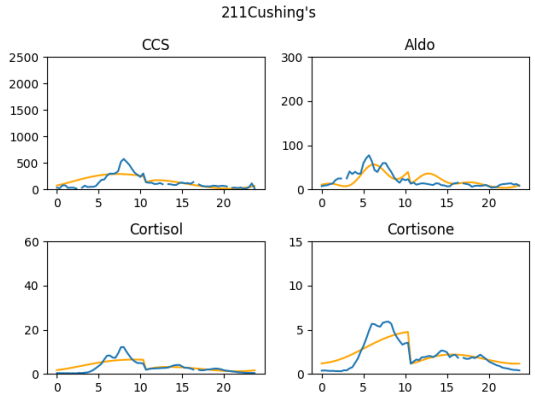
values of $Kappa_c$ and $Kappa_f$ than Cushing's.

The results of PirmAldo are not as good as Cushing's. The degree of separation in the point estimation maps is not distinct. Confusion tables of Control versus PrimAldo have come to the same conclusion.

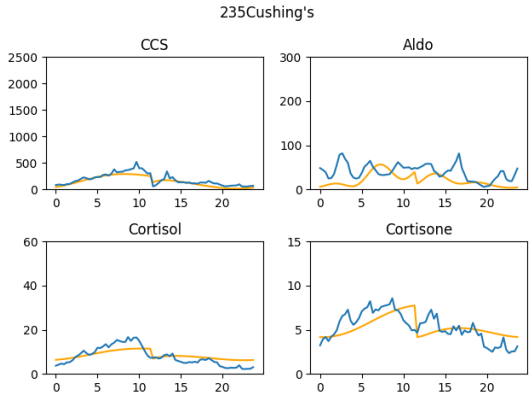
SVM Cushing's	True Control	True Cushing's
Predicted Control	0.83 +- 0.12	0.17 +- 0.12
Predicted Cushing's	0.05 +- 0.09	0.95 +- 0.09
Sensitivity	0.85	
Accuracy	0.89	

LR Cushing's	True Control	True Cushing's
Predicted Control	0.84 +- 0.11	0.16 +- 0.11
Predicted Cushing's	0.06 +- 0.10	0.94 +- 0.10
Sensitivity	0.85	
Accuracy	0.89	

Table 5.8: Confusion tables on parameter $Kappa_c$

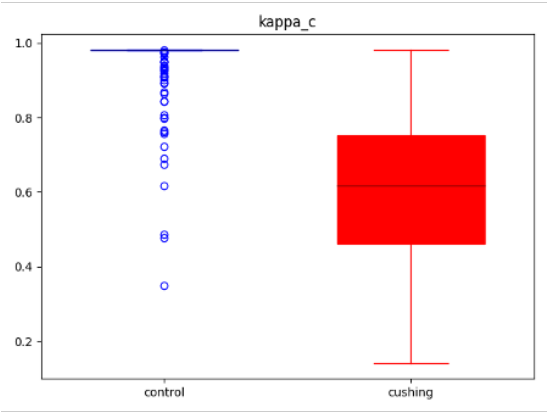


(a) SID:211 Cushing's

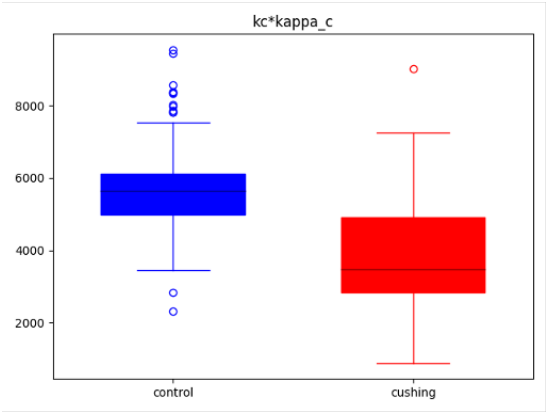


(b) SID:235 Cushing's

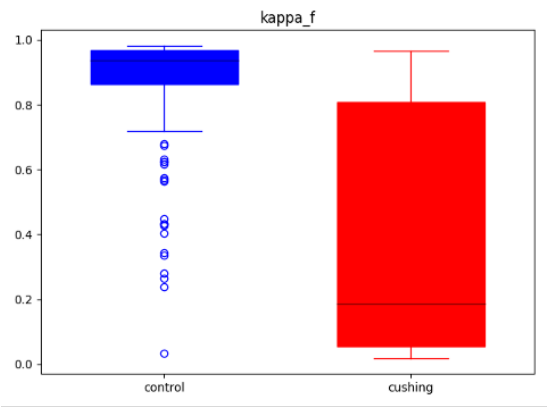
Figure 5.25: Trajectories of Cushing's subjects 211 and 235



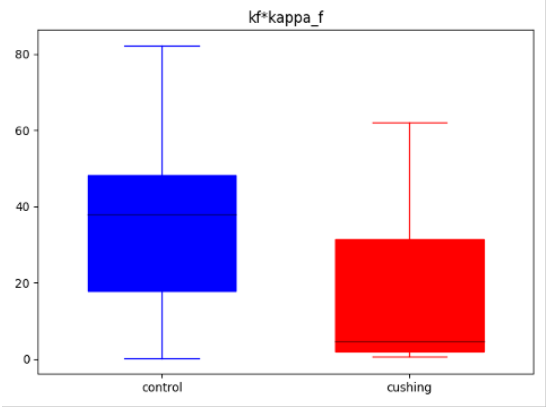
(a) $kappa_c$



(b) $k_c \cdot kappa_c$



(c) $kappa_f$



(d) $k_f \cdot kappa_f$

Figure 5.26: Box plots of important parameters

5.8 Summary

In this chapter, a time series classification approach in the model space has been proposed and has been compared with the classification method in the signal space. Both time series classifications in the model space and signal space have been demonstrated on a filtered real-world data set with 140 subjects in three steroid hormone biosynthesis conditions. Each subject is a multivariate time series with missing values and different starting times. During the data processing stage, the univariate Gaussian process has been used to impute missing values. Starting time issue has been solved by taking the first measurements at 5 pm and the last measurements at 10 am. In order to extract important features, a robust feature selection approach has been developed. Additionally, a detailed experiment methodology has been designed based on three degrees of design freedom. Then, two classifier models SVM and logistic regression have been trained in both model and signal space, with the full set and subset of features, using full and partial adrenal steroid pathway models respectively. Also, repeated undersampling and classifier ensembles have been used to deal with the class imbalance problem. Confusion tables and feature selection results of different classifiers have been analysed and evaluated. In the end, a hybrid model, which is a combination of classifiers in the model and signal space has been proposed to improve the overall classification performance. It has also been tested by eight additional Cushing's subjects with a 100% accuracy rate.

Moreover, the classification approach in the model space has also been demonstrated on an unfiltered data set, which contains 270 subjects in total. The classification in the

signal space has not been used because some subjects in this unfiltered data set involve large observational gaps, which the univariate Gaussian process is not able to impute. Fortunately, handling missing values is one of the advantages of the learning in the model space approach. Then, the adjoint method has been employed to handle the initial value problem for the ODE solver. All experiments have been conducted with the same design as before. A hybrid model has been proposed by combining two classifiers both in the model space. It has also been evaluated by those eight additional Cushing's subjects.

To conclude, classification in the model space not only can have comparable performance but also can handle complex data sets (time series with substantial missing values), compared to classic classification in signal space. In addition, learning in the model space approach can provide explanations and interpretations for important results and key findings by taking biomedical or mechanistic information into account.

Chapter 6 Conclusion and future work

6.1 Conclusion

Time series is a very common type of data existing everywhere in the real world. Some examples involve temperature values across some time periods, stock market prices and biomedical measurements, etc. With the development of the digital healthcare system and digital clinical measures, biomedical time series data are widely available in recent years. Hence, there is an increasing demand of the development of data mining and classification approaches in order to discern and discover some useful biomedical information for more accurate diagnosis, screening, and monitoring. This will be highly advantageous for healthcare as a whole. However, biomedical time series have some particular challenges than time series in other fields. First of all, most biomedical time series data sets just contain a very small number of subjects because the data collection requires plenty of time, resources, and effort. Undoubtedly, most deep learning models that are data hungry are not suitable for such data sets with a small size. Secondly, measurements of biomedical data obtained in real-life settings are normally quite noisy and/or sparse and with missing values. It is not feasible to simply apply classic machine learning algorithms such as logistic regression and support vector machines on raw time series data. In addition, in order to improve the diagnosis using machine learning models, the model interpretation,

which is the ability to explain results from a biomedical point of view and discover some potential biomedical insights has to be involved during model building. However, it is still a challenge for most existing machine learning models. Therefore, a LiMS approach has been proposed in this research. LiMS does all learning in the model space so it can handle sparse and noisy time series data and also provide appropriate mechanistic and biomedical model interpretations. LiMS approaches have been demonstrated on an adrenal steroid hormones data set, who has sparse and noisy measurements and a small number of subjects. In **Chapter 1**, a background introduction, research motivations, objectives, and contributions have been presented.

Time series data mining is a useful tool to discover hidden knowledge or information from either original or transformed time series data. Classification and clustering are two commonly used approaches in time series data mining. **Chapter 2** has reviewed relevant research on time series classification and clustering in the literature, providing the necessary knowledge and an overall picture for the following chapters. Common approaches of both time series classification and clustering have been discussed and reviewed, particularly self-organising maps and its probabilistic version generative topographic mapping. Some existing machine learning applications on multivariate steroid data have been discussed to have an overall deep understanding of machine learning in steroid data.

Chapter 3 has thoroughly discussed the adrenal steroid system to provide a general understanding of the adrenal steroid hormone data set that LiMS approaches have been demonstrated on. Key components (adrenal glands and the brain) and three main path-

ways together with their corresponding hormones (particularly cortisol, cortisone, aldosterone, and corticosterone) of the adrenal steroid system have been introduced in detail. Also, three adrenal disorders caused by the excess or insufficiency of certain hormones have been presented together with their causes and symptoms, etc.

Clustering is the most common method for time series pattern discovery. It can also help to visualize time series via some topographic techniques such as SOM and GTM, which is a probabilistic formulation of SOM. **Chapter 4** has presented a novel learning approach for SOM formulated in the model space called SOMiMS, following with an extension of GTM in the model space for clustering and visualizing sparse and noisy time series data. Both SOMiMS and extended GTM have been demonstrated on the adrenal steroid hormone data set. A parametric mechanistic inferential model has been formulated in the form of ordinary differential equations. Then, each time series data has been transformed into a vector of model parameters, which is the representative of the corresponding mechanistic model. Given transformed data, topographic maps have been generated in the space of those mechanistic models, providing the chance to readily interpret the topographic data organisation from the mechanistic point of view. In addition, parameter plots of topographic maps have been provided as heat maps showing values learnt of each individual mechanistic model parameter across nodes on the mapping. A KNN classification has also been used to evaluate and quantify the degree of separation of different conditions on the mapping plot. Compared to other classic approaches working in the signal space, SOMiMS and extended GTM of sparse and noisy time series data have shown the good performance of separation of different conditions. It has also taken mechanistic informa-

tion into account and created interpretable readily time series visualisations.

Besides topographic mapping, this thesis has also developed a classification approach in the model space applied on the filtered adrenal steroid data set. In the preprocessing stage, **Chapter 5** has employed the univariate Gaussian process to impute missing values for raw time series data set. In the feature generation step, raw data have been transferred into model space using MLE with constraints. Also, the adjoint method has been included to solve the initial value problem of the ODEs. Moreover, it has been found that the data set has a class imbalance problem, together with a small data set size issue. Repeated undersampling and classifier ensembles have been combined to handle these problems in the classification training stage. Apart from the classification in the model space, classification in the signal space of imputed time series data has also been provided as a benchmark. Hence, the feature selection has taken place in both the signal and model space. Features refer to time points in the signal space, while model parameters are in the model space. Important features have been selected based on importance plots associated with the coefficients of ensemble classifiers. All experiments have been designed based on the 3 degrees of freedom, which has been explained in **Chapter 5**. Both classifier models SVM and logistic regression have been applied to this task. In the end, a hybrid model, which is a combination of classification models in signal and model space has been proposed. It has also been demonstrated on the adrenal steroid data set with an impressive performance. Generally, although classification performance in the model space is similar to the signal space, classifiers and feature selection in the model space have been able to provide various mechanistic/biomedical insights and interpretations. Some surprising

biomedical phenomena have been highlighted in **Chapter 5**.For example, Cushing's which is caused by the excess of Cortisol can also be affected by CCS especially during the morning. Also, the classifier based solely on parameter $Kappa_f$ can predict Cushing's with a very high accuracy. In addition, classification of Control versus Cushing's in the subset feature space performs better than the full feature space, while classification of Control versus PrimAldo requires full features. Furthermore, the classification in the model space has also been demonstrated on the unfiltered adrenal steroid hormone data set, which contains large numbers of missing values with impressive performance.

6.2 Future work

Some possible future work has been summarized as follows:

- In **chapter 4**, SOMiMS and extended GTM have been demonstrated on the data set with three conditions: Control, Cushing's, and PrimAldo. Apart from the separation of three conditions, both maps have shown a tendency of sub-grouping the Cuhsing's cohort into sub-populations. Hence, applying SOMiMS and extended GTM to Cushing's profiles only would be helpful to the discovery of Cushing's subtype structure. Moreover, the data set used in **chapter 4** is filtered with 60 steroid profiles in total. It would be interesting to see topographic maps trained on the unfiltered data set.

- In this thesis, each time series has been transferred into a vector of model parameters of a suitable underlying mechanistic model. However, there might be more than one model that could well represent this data because the time series is very sparse and

noisy. As a result, it would be worth trying to represent each individual using all possible models, for example using posterior distribution over models in the future.

- Missing values of most biomedical time series data sets are unfortunately unavoidable because of some obstacles and difficulties faced during data collection. Missing value imputation is the most common way to deal with those missing values. The univariate Gaussian process has been used to impute missing values for the filtered data set in this research. However, it cannot be used to fill subjects with large observational gaps (e.g. whole metabolise missing). Then, it would be worth trying some other imputation techniques such as the multi-output Gaussian process.

- Instead of combining the two models (signal space and model space), it would be worth developing a model where both kind of features are concatenated or a model using features in the model space together with features representing the errors of the model in the signal space.

Bibliography

[1] Aileen Nielsen. *Practical time series analysis: Prediction with statistics and machine learning*. O'Reilly Media, 2019.

[2] Steven Cheng-Xian Li and Benjamin M Marlin. "Classification of Sparse and Irregularly Sampled Time Series with Mixtures of Expected Gaussian Kernels and Random Features." In: *UAI*. 2015, pp. 484–493.

[3] Benjamin M Marlin et al. "Unsupervised pattern discovery in electronic health care data using probabilistic clustering models". In: *Proceedings of the 2nd ACM SIGHIT international health informatics symposium*. 2012, pp. 389–398.

[4] T Ruf. "The Lomb-Scargle periodogram in biological rhythm research: analysis of incomplete and unequally spaced time-series". In: *Biological Rhythm Research* 30.2 (1999), pp. 178–201.

[5] Christian Bock et al. "Machine learning for biomedical time series classification: from shapelets to deep learning". In: *Artificial Neural Networks* (2021), pp. 33–71.

[6] Saeed Aghabozorgi, Ali Seyed Shirkhorshidi, and Teh Ying Wah. "Time-series clustering–a decade review". In: *Information systems* 53 (2015), pp. 16–38.

[7] R Pradeep Kumar and P Nagabhushan. "Time Series as a Point-A Novel Approach for Time Series Cluster Visualization." In: *DMIN*. Citeseer. 2006, pp. 24–29.

[8] Xiaozhe Wang et al. "A scalable method for time series clustering". In: *Unrefereed research papers* 1 (2004).

[9] Hui Zhang et al. "Unsupervised feature extraction for time series clustering using orthogonal wavelet transform". In: *Informatica* 30.3 (2006).

[10] Jessica Lin et al. "Iterative incremental clustering of time series". In: *Advances in Database Technology-EDBT 2004: 9th International Conference on Extending Database Technology, Heraklion, Crete, Greece, March 14-18, 2004 9*. Springer. 2004, pp. 106–122.

[11] Will Ke Wang et al. "A Systematic Review of Time Series Classification Techniques Used in Biomedical Applications". In: *Sensors* 22.20 (2022), p. 8016.

[12] Hassan Ismail Fawaz et al. "Deep learning for time series classification: a review". In: *Data mining and knowledge discovery* 33.4 (2019), pp. 917–963.

[13] Nguyen Xuan Anh, Ramesh Mark Nataraja, and Sunita Chauhan. "Towards near real-time assessment of surgical skills: A comparison of feature extraction techniques". In: *Computer methods and programs in biomedicine* 187 (2020), p. 105234.

[14] SS Sreeja Mole and K Sujatha. "An efficient Gait Dynamics classification method for Neurodegenerative Diseases using Brain signals". In: *Journal of medical systems* 43.8 (2019), p. 245.

[15] Yousef Rezaei Tabar et al. "Investigation of low dimensional feature spaces for automatic sleep staging". In: *Computer Methods and Programs in Biomedicine* 205 (2021), p. 106091.

[16] Beatriz Garcia-Martinez et al. "Nonlinear predictability analysis of brain dynamics for automatic recognition of negative stress". In: *Neural Computing and Applications* 32 (2020), pp. 13221–13231.

[17] Yanjun Li et al. "Probability density distribution of delta RR intervals: a novel method for the detection of atrial fibrillation". In: *Australasian physical & engineering sciences in medicine* 40 (2017), pp. 707–716.

[18] Jonathan G Richens, Ciarán M Lee, and Saurabh Johri. "Improving the accuracy of medical diagnosis with causal machine learning". In: *Nature communications* 11.1 (2020), p. 3923.

[19] Xiaozhe Wang, Kate Smith, and Rob Hyndman. "Characteristic-based clustering for time series data". In: *Data mining and knowledge Discovery* 13 (2006), pp. 335–364.

[20] Teuvo Kohonen. "The self-organizing map". In: *Proceedings of the IEEE* 78.9 (1990), pp. 1464–1480.

[21] Christopher M Bishop, Markus Svensén, and Christopher KI Williams. "GTM: The generative topographic mapping". In: *Neural computation* 10.1 (1998), pp. 215–234.

[22] Zhengzheng Xing, Jian Pei, and Eamonn Keogh. "A brief survey on sequence classification". In: *ACM Sigkdd Explorations Newsletter* 12.1 (2010), pp. 40–48.

[23] Neal Lesh, Mohammed J Zaki, and Mitsunori Ogihara. "Mining features for sequence classification". In: *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*. 1999, pp. 342–346.

[24] Charu C Aggarwal. "On effective classification of strings with wavelets". In: *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and Data Mining*. 2002, pp. 163–172.

[25] Eamonn J Keogh and Michael J Pazzani. "Scaling up dynamic time warping for datamining applications". In: *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*. 2000, pp. 285–289.

[26] C Lee Giles, Steve Lawrence, and Ah Chung Tsoi. "Noisy time series prediction using recurrent neural networks and grammatical inference". In: *Machine learning* 44.1-2 (2001), p. 161.

[27] Prashant K Srivastava et al. "HMM-ModE–Improved classification using profile hidden Markov models by optimising the discrimination threshold and modifying emission probabilities with negative training sequences". In: *BMC bioinformatics* 8.1 (2007), pp. 1–17.

[28] Lawrence R Rabiner. "A tutorial on hidden Markov models and selected applications in speech recognition". In: *Proceedings of the IEEE* 77.2 (1989), pp. 257–286.

[29] Yuan Shen, Peter Tino, and Krasimira Tsaneva-Atanasova. "Classification framework for partially observed dynamical systems". In: *Physical Review E* 95.4 (2017), p. 043303.

[30] Huanhuan Chen et al. "Model-based kernel for efficient time series analysis". In: *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2013, pp. 392–400.

[31] Fabio Cuzzolin. "Manifold learning for multi-dimensional auto-regressive dynamical models". In: *Machine Learning for Vision-Based Motion Analysis*. Springer, 2011, pp. 55–74.

[32] Fabio Cuzzolin and Michael Sapienza. "Learning pullback HMM distances". In: *IEEE transactions on pattern analysis and machine intelligence* 36.7 (2013), pp. 1483–1489.

[33] Pedro Moreno, Purdy Ho, and Nuno Vasconcelos. "A Kullback-Leibler divergence based kernel for SVM classification in multimedia applications". In: *Advances in neural information processing systems* 16 (2003).

[34] Antoni B Chan and Nuno Vasconcelos. "Probabilistic kernels for the classification of auto-regressive visual processes". In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. Vol. 1. IEEE. 2005, pp. 846–851.

[35] Tony Jebara, Risi Kondor, and Andrew Howard. "Probability product kernels". In: *The Journal of Machine Learning Research* 5 (2004), pp. 819–844.

[36] SVN Vishwanathan, Alexander J Smola, and René Vidal. "Binet-Cauchy kernels on dynamical systems and its application to the analysis of dynamic scenes". In: *International Journal of Computer Vision* 73.1 (2007), pp. 95–119.

[37] Tommi Jaakkola and David Haussler. "Exploiting generative models in discriminative classifiers". In: *Advances in neural information processing systems* 11 (1998).

[38] Marco Cuturi and Arnaud Doucet. "Autoregressive kernels for time series". In: *arXiv preprint arXiv:1101.0673* (2011).

[39] Pradeep Rai and Shubha Singh. "A survey of clustering techniques". In: *International Journal of Computer Applications* 7.12 (2010), pp. 1–5.

[40] J MacQueen. "Classification and analysis of multivariate observations". In: *5th Berkeley Symp. Math. Statist. Probability*. University of California Los Angeles LA USA. 1967, pp. 281–297.

[41] Leonard Kaufman and Peter J Rousseeuw. *Finding groups in data: an introduction to cluster analysis*. John Wiley & Sons, 2009.

[42] Philippe Esling and Carlos Agon. "Time-series data mining". In: *ACM Computing Surveys (CSUR)* 45.1 (2012), pp. 1–34.

[43] Ville Hautamaki, Pekka Nykanen, and Pasi Franti. "Time-series clustering by approximate prototypes". In: *2008 19th International conference on pattern recognition*. IEEE. 2008, pp. 1–4.

[44] Chonghui Guo, Hongfeng Jia, and Na Zhang. "Time series clustering based on ICA for stock data analysis". In: *2008 4th international conference on wireless communications, networking and mobile computing*. IEEE. 2008, pp. 1–4.

[45] James C Bezdek. *Pattern recognition with fuzzy objective function algorithms*. Springer Science & Business Media, 2013.

[46] Doulaye Dembele and Philippe Kastner. "Fuzzy C-means method for clustering microarray data". In: *bioinformatics* 19.8 (2003), pp. 973–980.

[47] Jonathan Alon et al. "Discovering clusters in motion time-series data". In: *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.* Vol. 1. IEEE. 2003, pp. I–I.

[48] Xavier Golay et al. "A new correlation-based fuzzy logic clustering algorithm for FMRI". In: *Magnetic resonance in medicine* 40.2 (1998), pp. 249–260.

[49] Dat Tran and Michael Wagner. "Fuzzy c-means clustering-based speaker verification". In: *Advances in Soft Computing—AFSS 2002: 2002 AFSS International Conference on Fuzzy Systems Calcutta, India, February 3–6, 2002 Proceedings*. Springer. 2002, pp. 318–324.

[50] Tim Oates, Matthew D Schmill, and Paul R Cohen. "A method for clustering the experiences of a mobile robot that accords with human judgments". In: *AAAI/IAAI*. 2000, pp. 846–851.

[51] Martin Ester et al. "A density-based algorithm for discovering clusters in large spatial databases with noise." In: *kdd*. Vol. 96. 34. 1996, pp. 226–231.

[52] Mihael Ankerst et al. "OPTICS: Ordering points to identify the clustering structure". In: *ACM Sigmod record* 28.2 (1999), pp. 49–60.

[53] Teuvo Kohonen. "Essentials of the self-organizing map". In: *Neural networks* 37 (2013), pp. 52–65.

[54] Christophe Biernacki, Gilles Celeux, and Gérard Govaert. "Assessing a mixture model for clustering with the integrated completed likelihood". In: *IEEE transactions on pattern analysis and machine intelligence* 22.7 (2000), pp. 719–725.

[55] Anthony Bagnall and Gareth Janacek. "Clustering time series with clipped data". In: *Machine learning* 58 (2005), pp. 151–178.

[56] Manuele Bicego, Vittorio Murino, and Mário AT Figueiredo. "Similarity-based clustering of sequences using hidden Markov models". In: *Machine Learning and Data Mining in Pattern Recognition: Third International Conference, MLDM 2003 Leipzig, Germany, July 5–7, 2003 Proceedings 3*. Springer. 2003, pp. 86–95.

[57] He Ni and Hujun Yin. "A self-organising mixture autoregressive network for FX time series modelling and prediction". In: *Neurocomputing* 72.16-18 (2009), pp. 3529–3537.

[58] Peter Tino, Ata Kabán, and Yi Sun. "A generative probabilistic approach to visualizing sets of symbolic sequences". In: *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. 2004, pp. 701–706.

[59] Nikolaos Gianniotis and Peter Tino. "Visualization of tree-structured data through generative topographic mapping". In: *IEEE Transactions on Neural Networks* 19.8 (2008), pp. 1468–1493.

[60] Edmund H Wilkes, Gill Rumsby, and Gary M Woodward. "Using machine learning to aid the interpretation of urine steroid profiles". In: *Clinical chemistry* 64.11 (2018), pp. 1586–1595.

[61] Graeme Eisenhofer et al. "Use of steroid profiling combined with machine learning for identification and subtype classification in primary aldosteronism". In: *JAMA Network Open* 3.9 (2020), e2016209–e2016209.

[62] Vasileios Chortis et al. "Urine steroid metabolomics as a novel tool for detection of recurrent adrenocortical carcinoma". In: *The Journal of Clinical Endocrinology & Metabolism* 105.3 (2020), e307–e318.

[63] Jiliang Tang, Salem Alelyani, and Huan Liu. "Feature selection for classification: A review". In: *Data classification: Algorithms and applications* (2014), p. 37.

[64] Saroj Nimkarn and Maria I New. "Disorders of the Adrenal Gland". In: *Avery's Diseases of the Newborn*. Elsevier, 2012, pp. 1274–1285.

[65] Gabriela Paula Finkielstain, Smita Jha, and Deborah Merke. "Adrenal disorders". In: *Biochemical and Molecular Basis of Pediatric Disease*. Elsevier, 2021, pp. 267–296.

[66] Meghan Dutt, Chase J Wehrle, and Ishwarlal Jialal. "Physiology, adrenal gland". In: *StatPearls [Internet]*. StatPearls Publishing, 2021.

[67] Shinji Nomura et al. "Clinical significance of cortisone and cortisone/cortisol ratio in evaluating children with adrenal diseases". In: *Clinica chimica acta* 256.1 (1996), pp. 1–11.

[68] Paul Stewart and William Young. "Primary Aldosteronism". In: *The Journal of Clinical Endocrinology & Metabolism* 92.12 (2007), E1–E1.

[69] Walter L Miller et al. "The adrenal cortex and its disorders". In: *Sperling Pediatric Endocrinology*. Elsevier, 2021, pp. 425–490.

[70] Markus Torma. "Kohonen self-organizing feature map and its use in clustering". In: *ISPRS Commission III Symposium: Spatial Information from Digital Photogrammetry and Computer Vision*. Vol. 2357. SPIE. 1994, pp. 830–835.

[71] Wiebke Arlt and Paul M Stewart. "Adrenal corticosteroid biosynthesis, metabolism, and action". In: *Endocrinology and Metabolism Clinics* 34.2 (2005), pp. 293–313.

[72] Wiebke Arlt et al. "Steroid metabolome analysis reveals prevalent glucocorticoid excess in primary aldosteronism". In: *JCI insight* 2.8 (2017).

[73] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, 2006, pp. I–XVIII, 1–248. ISBN: 026218253X.

[74] James M Keller, Michael R Gray, and James A Givens. "A fuzzy k-nearest neighbor algorithm". In: *IEEE transactions on systems, man, and cybernetics* 4 (1985), pp. 580–585.

[75] Carl Edward Rasmussen. "Gaussian processes in machine learning". In: *Advanced Lectures on Machine Learning: ML Summer Schools 2003, Canberra, Australia, February 2-14, 2003, Tübingen, Germany, August 4-16, 2003, Revised Lectures*. Springer, 2004, pp. 63–71.

[76] Ian Nabney. *NETLAB: algorithms for pattern recognition*. Springer Science & Business Media, 2002.

[77] In Jae Myung. "Tutorial on maximum likelihood estimation". In: *Journal of mathematical Psychology* 47.1 (2003), pp. 90–100.

[78] Corinna Cortes and Vladimir Vapnik. "Support-vector networks". In: *Machine learning* 20.3 (1995), pp. 273–297.

[79] Nello Cristianini and John Shawe-Taylor. "Support Vector Machines". In: *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2000, pp. 93–124. DOI: 10.1017/CBO9780511801389.008.

[80] Dustin Boswell. "Introduction to support vector machines". In: *Departement of Computer Science and Engineering University of California San Diego* (2002).

[81] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[82] Stephan Dreiseitl and Lucila Ohno-Machado. "Logistic regression and artificial neural network classification models: a methodology review". In: *Journal of biomedical informatics* 35.5-6 (2002), pp. 352–359.

[83] Che Junfei, Wu Qingfeng, and Dong Huailin. "An empirical study on ensemble selection for class-imbalance data sets". In: *2010 5th International Conference on Computer Science & Education*. IEEE. 2010, pp. 477–480.

[84] Leo Breiman. "Random forests". In: *Machine learning* 45.1 (2001), pp. 5–32.

[85] Xu-Ying Liu, Jianxin Wu, and Zhi-Hua Zhou. "Exploratory undersampling for class-imbalance learning". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 39.2 (2008), pp. 539–550.

[86] Roweida Mohammed, Jumanah Rawashdeh, and Malak Abdullah. "Machine learning with oversampling and undersampling techniques: overview study and experimental results". In: *2020 11th international conference on information and communication systems (ICICS)*. IEEE. 2020, pp. 243–248.

[87] Joseph L Schafer. "Multiple imputation: a primer". In: *Statistical methods in medical research* 8.1 (1999), pp. 3–15.

[88] Andrew M. Bradley. *PDE-constrained optimization and the adjoint method*. Oct. 2019. URL: https://cs.stanford.edu/~ambrad/adjoint_tutorial.pdf.

Appendix A Appendix

A.1 Filtered data set in the model space with missing value imputation

A.1.1 Model space full steroid pathway model

Feature space - full space

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.80 +- 0.12	0.20 +- 0.12
Predicted Cushing's	0.07 +- 0.07	0.93 +- 0.07
Sensitivity	0.82	
Accuracy	0.87	

LR Cushing's	True Control	True Cushing's
Predicted Control	0.78 +- 0.17	0.22 +- 0.17
Predicted Cushing's	0.13 +- 0.15	0.87 +- 0.15
Sensitivity	0.80	
Accuracy	0.83	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.80 +- 0.14	0.20 +- 0.14
Predicted PrimAldo	0.19 +- 0.14	0.81 +- 0.14
Sensitivity	0.80	
Accuracy	0.81	

LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.78 +- 0.17	0.22 +- 0.17
Predicted PrimAldo	0.17 +- 0.16	0.83 +- 0.16
Sensitivity	0.79	
Accuracy	0.81	

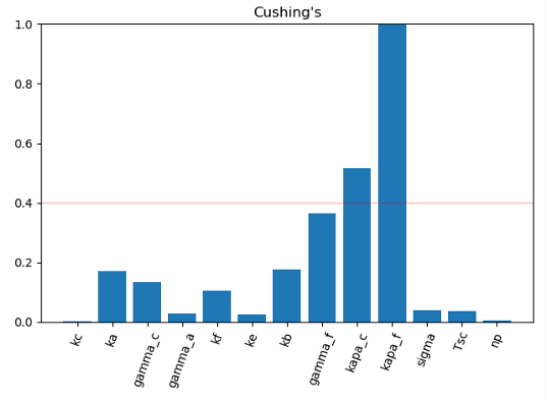
Feature space - subspace

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.83 +- 0.1	0.17 +- 0.1
Predicted Cushing's	0.03 +- 0.06	0.97 +- 0.06
Sensitivity	0.85	
Accuracy	0.90	

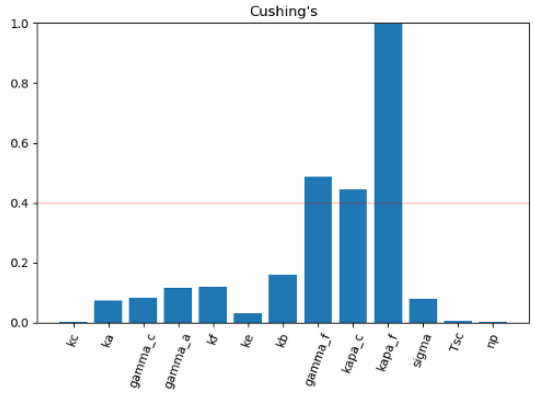
LR Cushing's	True Control	True Cushing's
Predicted Control	0.83 +- 0.16	0.17 +- 0.16
Predicted Cushing's	0.09 +- 0.12	0.91 +- 0.12
Sensitivity	0.84	
Accuracy	0.87	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.79 +- 0.14	0.21 +- 0.14
Predicted PrimAldo	0.19 +- 0.14	0.81 +- 0.14
Sensitivity	0.79	
Accuracy	0.80	

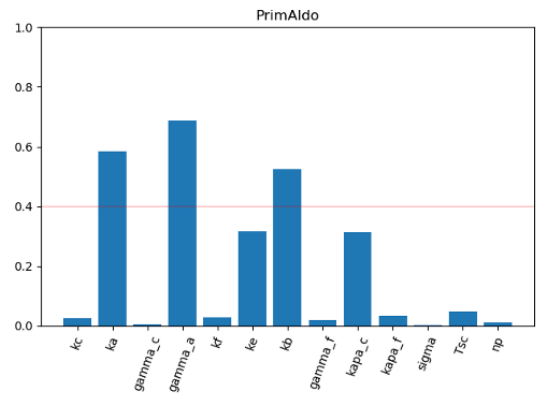
LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.80 +- 0.15	0.20 +- 0.15
Predicted PrimAldo	0.17 +- 0.16	0.83 +- 0.16
Sensitivity	0.81	
Accuracy	0.82	



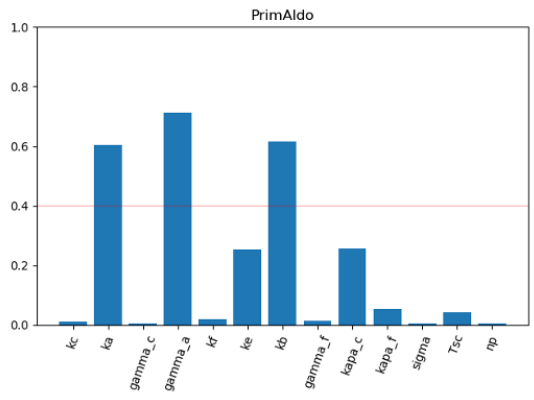
(a) svm model Cushing's full



(b) lr model Cushing's full



(c) svm model PrimAldo full



(d) lr model PrimAldo full

Figure A.1: Importance plots in the model space with full steroid pathway model

A.1.2 Model space partial steroid pathway model

Feature space - full space

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.82 +- 0.12	0.18 +- 0.12
Predicted Cushing's	0.05 +- 0.1	0.95 +- 0.1
Sensitivity	0.84	
Accuracy	0.89	

LR Cushing's	True Control	True Cushing's
Predicted Control	0.73 +- 0.18	0.27 +- 0.18
Predicted Cushing's	0.05 +- 0.09	0.95 +- 0.09
Sensitivity	0.78	
Accuracy	0.84	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.72 +- 0.13	0.28 +- 0.13
Predicted PrimAldo	0.28 +- 0.13	0.72 +- 0.13
Sensitivity	0.72	
Accuracy	0.72	

LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.70 +- 0.18	0.30 +- 0.18
Predicted PrimAldo	0.24 +- 0.19	0.76 +- 0.19
Sensitivity	0.72	
Accuracy	0.73	

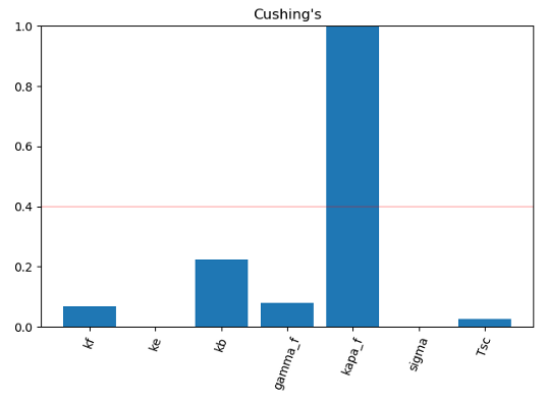
Feature space - subspace

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.81 +- 0.1	0.19 +- 0.1
Predicted Cushing's	0.02 +- 0.06	0.98 +- 0.06
Sensitivity	0.84	
Accuracy	0.90	

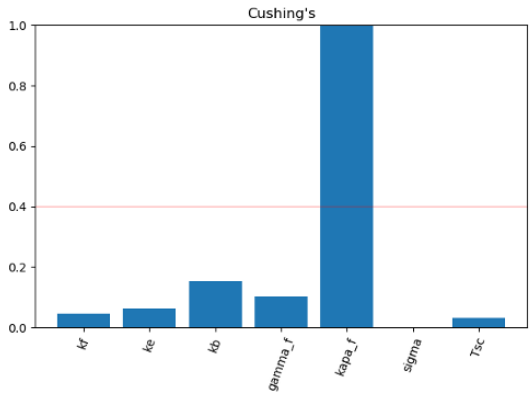
LR Cushing's	True Control	True Cushing's
Predicted Control	0.77 +- 0.17	0.23 +- 0.17
Predicted Cushing's	0.01 +- 0.04	0.99 +- 0.04
Sensitivity	0.76	
Accuracy	0.88	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.80 +- 0.16	0.20 +- 0.16
Predicted PrimAldo	0.23 +- 0.15	0.77 +- 0.15
Sensitivity	0.79	
Accuracy	0.79	

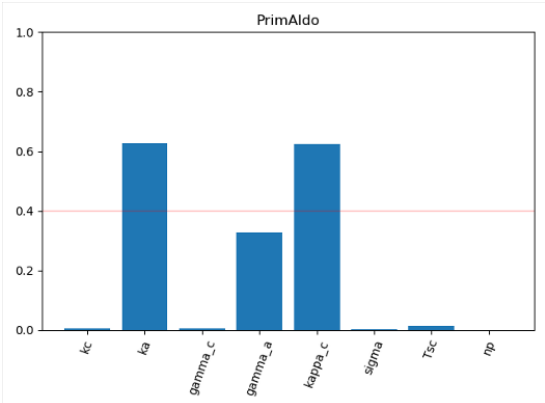
LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.43 +- 0.18	0.57 +- 0.18
Predicted PrimAldo	0.24 +- 0.17	0.76 +- 0.17
Sensitivity	0.57	
Accuracy	0.60	



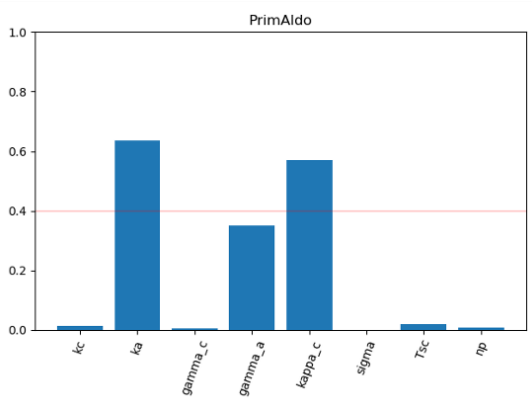
(a) svm model Cushing's partial



(b) lr model Cushing's partial



(c) svm model PrimAldo partial



(d) lr model PrimAldo partial

Figure A.2: Importance plots in the model space with partial steroid pathway model

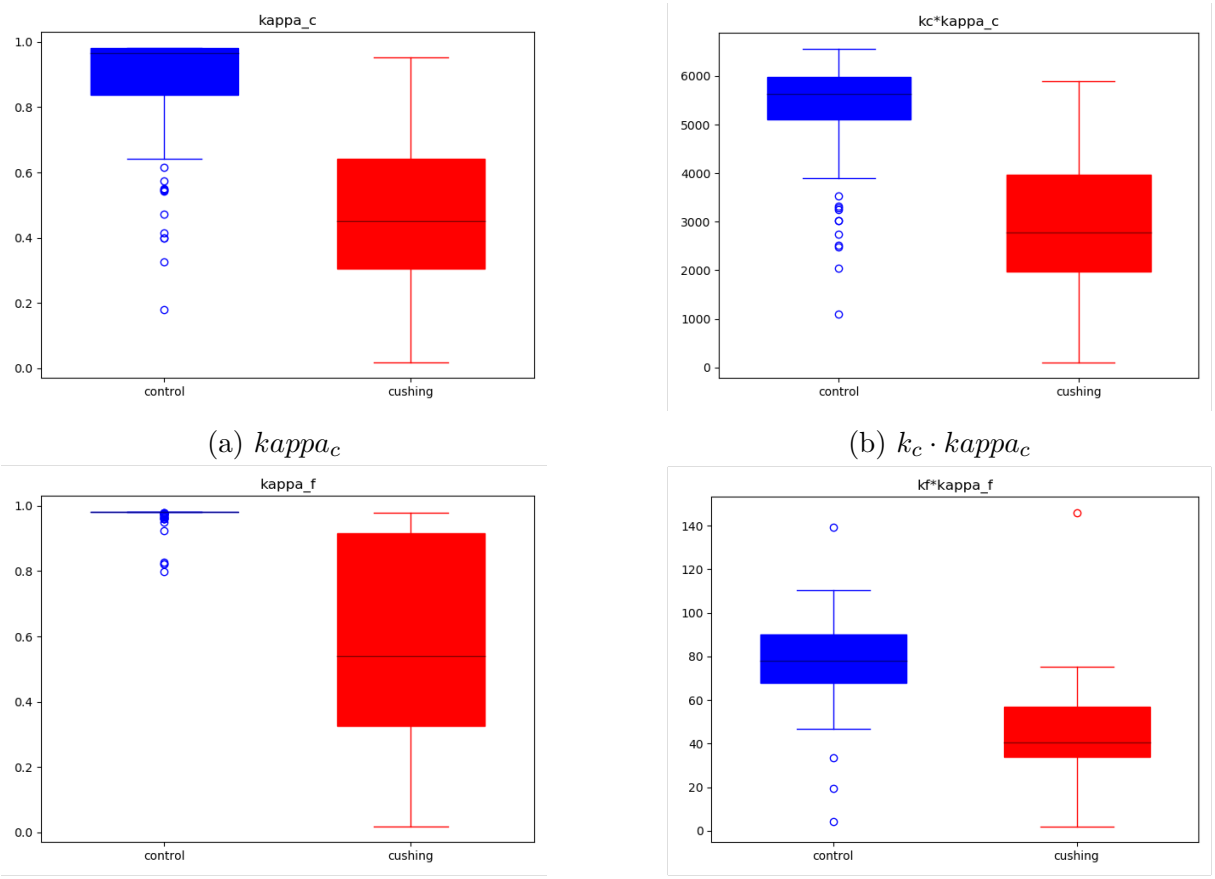
Hybrid

The hybrid model is developed by combining two base classifiers from the model space and signal space respectively. The classifier used from the model space is the one trained only on $kappa_f$. The classifier picked from the signal space is the one trained using full adrenal steroid pathway model in the subset feature space.

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.97 +- 0.07	0.03 +- 0.07
Predicted Cushing's	0.09 +- 0.11	0.91 +- 0.11
Sensitivity	0.97	
Accuracy	0.94	

LR Cushing's	True Control	True Cushing's
Predicted Control	0.97 +- 0.07	0.03 +- 0.07
Predicted Cushing's	0.07 +- 0.1	0.93 +- 0.1
Sensitivity	0.97	
Accuracy	0.95	

A.1.3 Point estimates in the model space



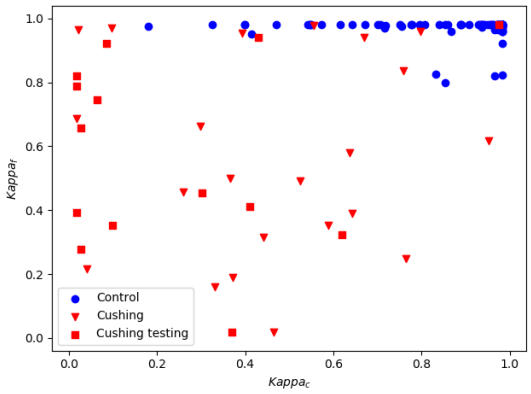
(a) $kappa_c$

(b) $k_c \cdot kappa_c$

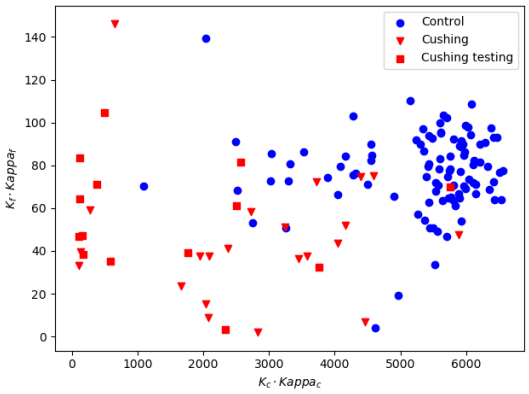
(c) $kappa_f$

(d) $k_f \cdot kappa_f$

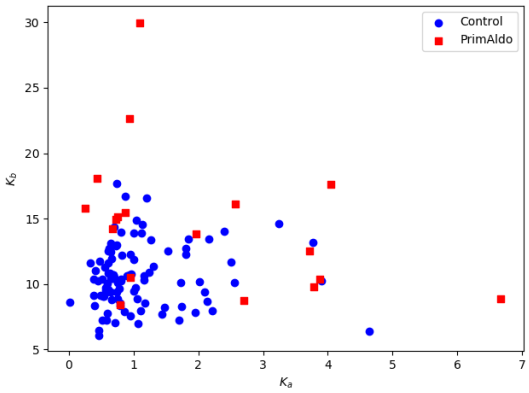
Figure A.3: Box plots of important parameters (with imputation)



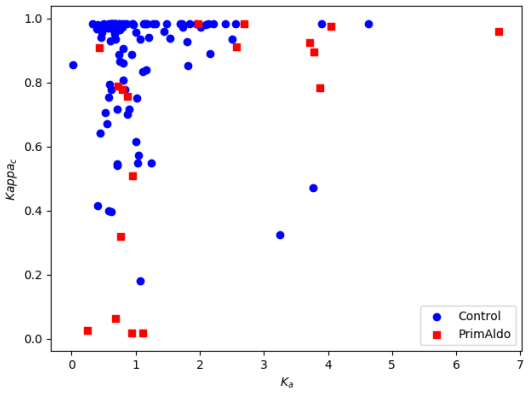
(a) $kappa_c$ and $kappa_f$



(b) $k_c \cdot kappa_c$ and $k_f \cdot kappa_f$



(c) k_a and k_b



(d) k_a and $kappa_c$

Figure A.4: Maps of point estimates (with imputation)

Appendix B Appendix

B.1 Classification results with threshold 0.6

B.1.1 Signal space

Signal space full adrenal steroid pathway model

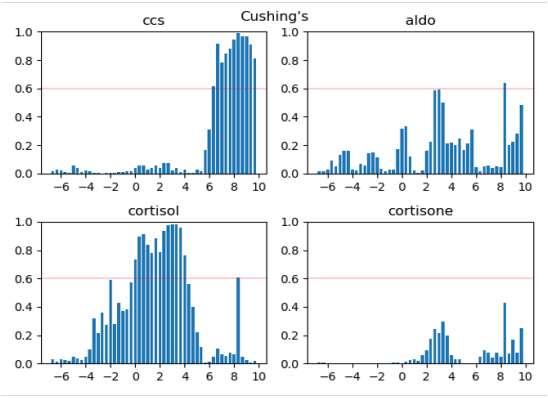
Feature space - subspace

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.92 +- 0.1	0.08 +- 0.1
Predicted Cushing's	0.07 +- 0.07	0.93 +- 0.07
Sensitivity	0.92	
Accuracy	0.93	

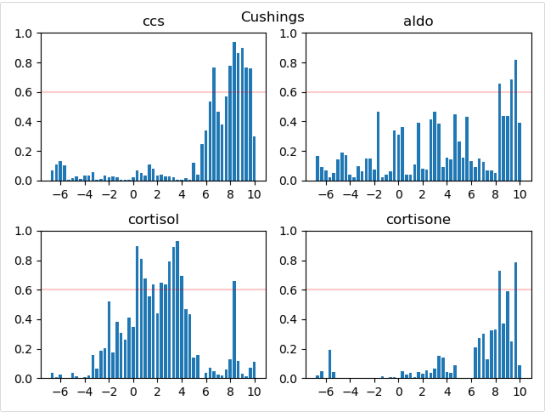
LR Cushing's	True Control	True Cushing's
Predicted Control	0.87 +- 0.14	0.13 +- 0.14
Predicted Cushing's	0.07 +- 0.09	0.93 +- 0.09
Sensitivity	0.88	
Accuracy	0.90	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.91 +- 0.12	0.09 +- 0.12
Predicted PrimAldo	0.18 +- 0.13	0.82 +- 0.13
Sensitivity	0.90	
Accuracy	0.87	

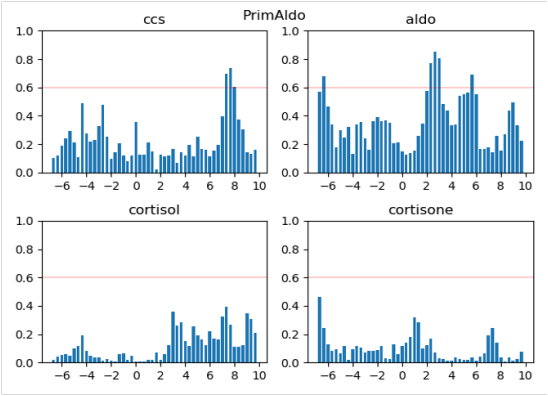
LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.82 +- 0.15	0.18 +- 0.15
Predicted PrimAldo	0.23 +- 0.19	0.77 +- 0.19
Sensitivity	0.81	
Accuracy	0.80	



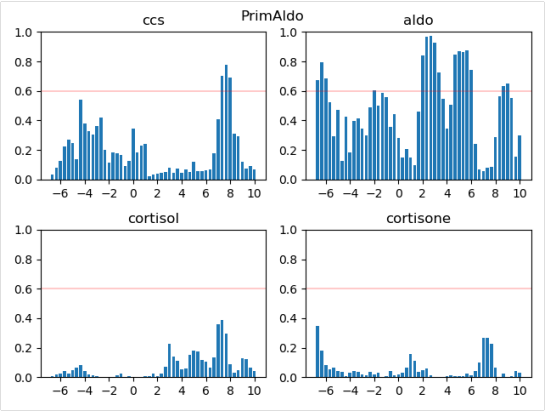
(a) svm signal Cushing's full



(b) lr signal Cushing's full



(c) svm signal PrimAldo full



(d) lr signal PrimAldo full

Figure B.1: Importance plots in the signal space with full adrenal steroid pathway model using threshold 0.6

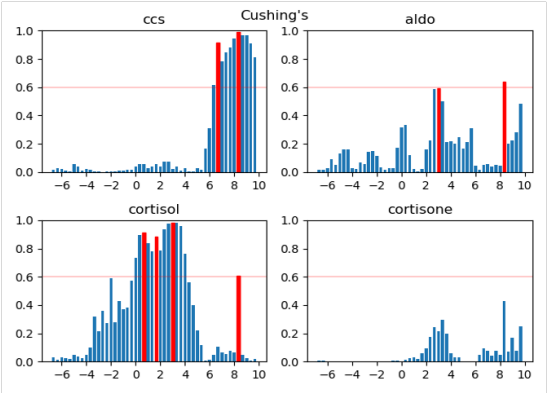
Feature space - peaks

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.89 +- 0.11	0.11 +- 0.11
Predicted Cushing's	0.05 +- 0.07	0.95 +- 0.07
Sensitivity	0.90	
Accuracy	0.92	

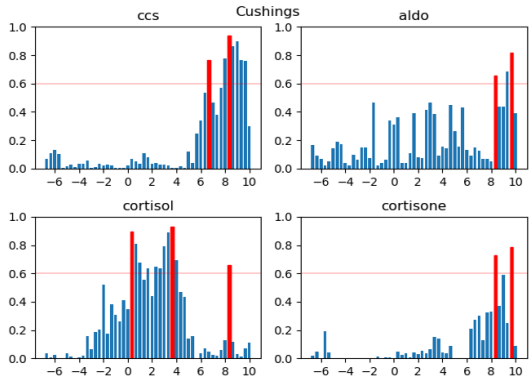
LR Cushing's	True Control	True Cushing's
Predicted Control	0.88 +- 0.13	0.12 +- 0.13
Predicted Cushing's	0.06 +- 0.09	0.94 +- 0.09
Sensitivity	0.89	
Accuracy	0.91	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.82 +- 0.13	0.18 +- 0.13
Predicted PrimAldo	0.16 +- 0.14	0.84 +- 0.14
Sensitivity	0.82	
Accuracy	0.83	

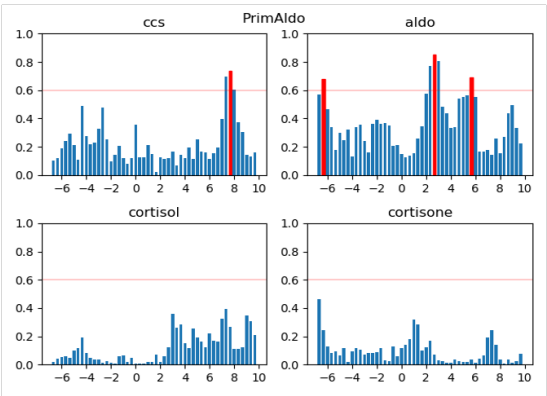
LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.83 +- 0.18	0.17 +- 0.18
Predicted PrimAldo	0.21 +- 0.17	0.79 +- 0.17
Sensitivity	0.82	
Accuracy	0.81	



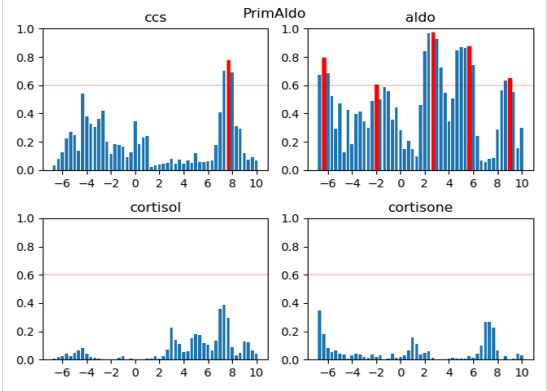
(a) svm signal Cushing's full on selected peaks



(b) lr signal Cushing's full on selected peaks



(c) svm signal PrimAldo full on selected peaks



(d) lr signal PrimAldo full on selected peaks

Figure B.2: Importance plots in the signal space with full adrenal steroid pathway model on selected peaks using threshold 0.6

Signal space partial adrenal steroid pathway model

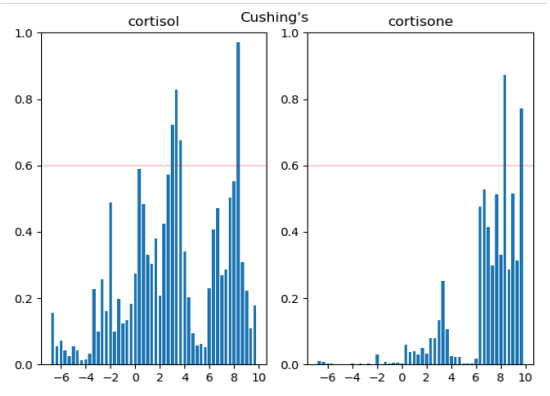
Feature space - subspace

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.77 +- 0.12	0.23 +- 0.12
Predicted Cushing's	0.08 +- 0.12	0.92 +- 0.12
Sensitivity	0.80	
Accuracy	0.85	

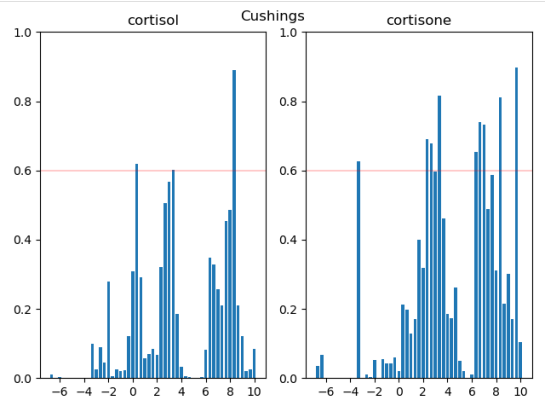
LR Cushing's	True Control	True Cushing's
Predicted Control	0.86 +- 0.14	0.14 +- 0.14
Predicted Cushing's	0.07 +- 0.12	0.93 +- 0.12
Sensitivity	0.87	
Accuracy	0.90	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.72 +- 0.12	0.28 +- 0.12
Predicted PrimAldo	0.16 +- 0.16	0.84 +- 0.16
Sensitivity	0.75	
Accuracy	0.78	

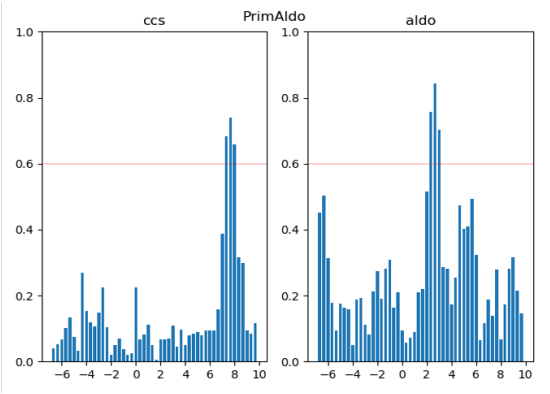
LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.75 +- 0.18	0.25 +- 0.18
Predicted PrimAldo	0.19 +- 0.17	0.81 +- 0.17
Sensitivity	0.76	
Accuracy	0.78	



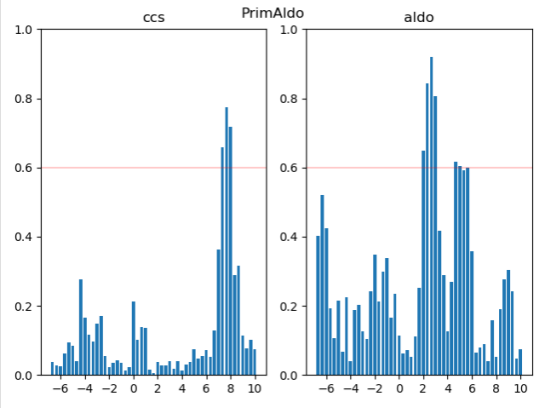
(a) svm signal Cushing's partial



(b) lr signal Cushing's partial



(c) svm signal PrimAldo partial



(d) lr signal PrimAldo partial

Figure B.3: Importance plots in the signal space with partial adrenal steroid pathway model using threshold 0.6

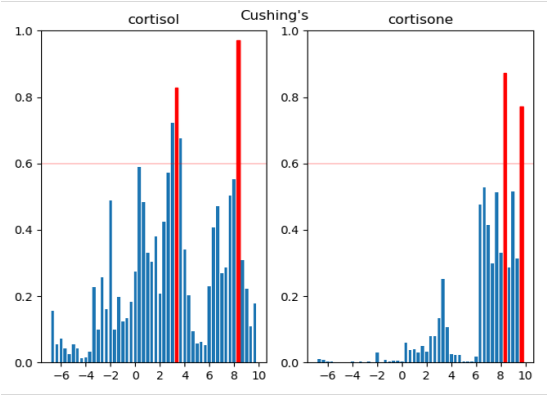
Feature space - peaks

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.75 +- 0.11	0.25 +- 0.11
Predicted Cushing's	0.08 +- 0.12	0.92 +- 0.12
Sensitivity	0.79	
Accuracy	0.84	

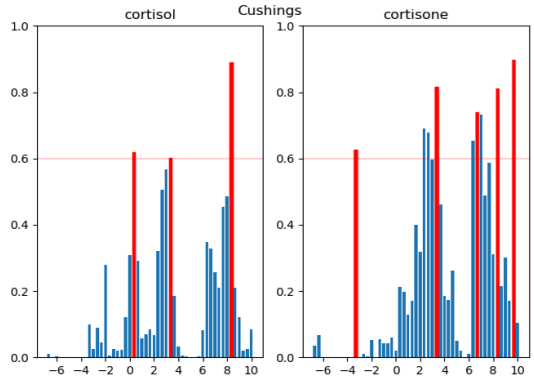
LR Cushing's	True Control	True Cushing's
Predicted Control	0.83 +- 0.14	0.17 +- 0.14
Predicted Cushing's	0.05 +- 0.08	0.95 +- 0.08
Sensitivity	0.85	
Accuracy	0.89	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.72 +- 0.12	0.28 +- 0.12
Predicted PrimAldo	0.18 +- 0.16	0.82 +- 0.16
Sensitivity	0.75	
Accuracy	0.77	

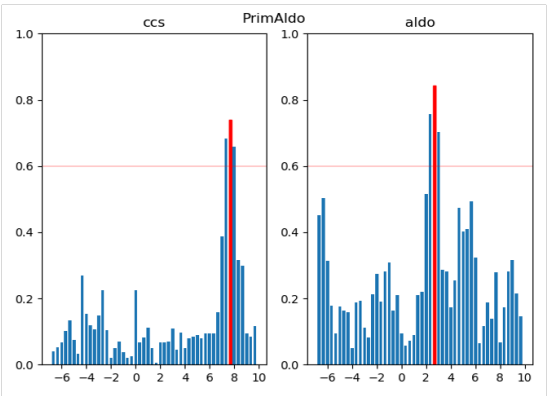
LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.71 +- 0.21	0.29 +- 0.21
Predicted PrimAldo	0.20 +- 0.17	0.80 +- 0.17
Sensitivity	0.73	
Accuracy	0.76	



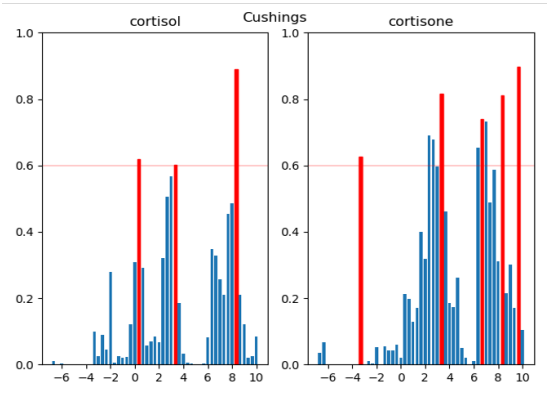
(a) svm signal Cushing's partial on selected peaks



(b) lr signal Cushing's partial on selected peaks



(c) svm signal PrimAldo partial on selected peaks



(d) lr signal PrimAldo partial on selected peaks

Figure B.4: Importance plots in the signal space with partial adrenal steroid pathway model on selected peaks using threshold 0.6

B.1.2 Model space

Model space full adrenal steroid pathway model

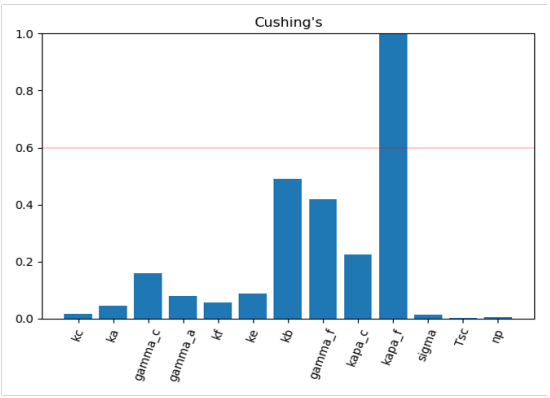
Feature space - subspace

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.81 +- 0.1	0.19 +- 0.1
Predicted Cushing's	0.02 +- 0.06	0.98 +- 0.06
Sensitivity	0.84	
Accuracy	0.90	

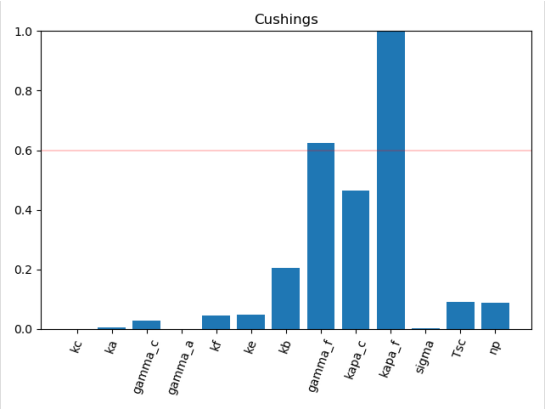
LR Cushing's	True Control	True Cushing's
Predicted Control	0.86 +- 0.16	0.14 +- 0.16
Predicted Cushing's	0.09 +- 0.13	0.91 +- 0.13
Sensitivity	0.87	
Accuracy	0.89	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.79 +- 0.13	0.21 +- 0.13
Predicted PrimAldo	0.20 +- 0.14	0.80 +- 0.14
Sensitivity	0.79	
Accuracy	0.89	

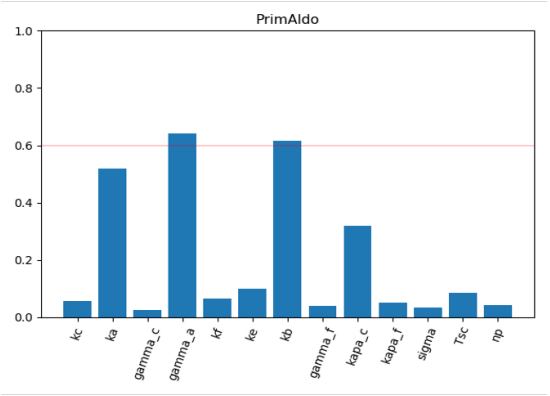
LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.70 +- 0.19	0.30 +- 0.19
Predicted PrimAldo	0.22 +- 0.2	0.78 +- 0.2
Sensitivity	0.72	
Accuracy	0.74	



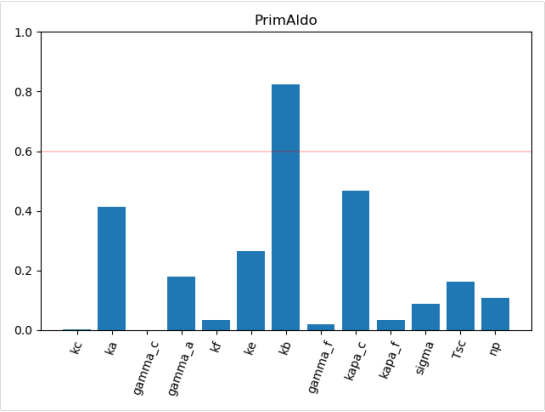
(a) svm model Cushing's full



(b) lr model Cushing's full



(c) svm model PrimAldo full



(d) lr model PrimAldo full

Figure B.5: Importance plots in the model space with full adrenal steroid pathway model using threshold 0.6

Model space partial adrenal steroid pathway model

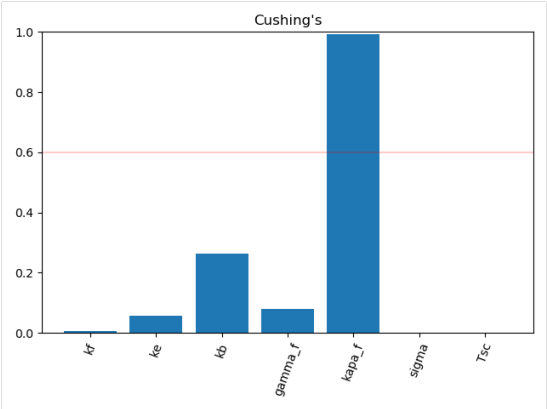
Feature space - subspace

SVM Cushing's	True Control	True Cushing's
Predicted Control	0.81 +- 0.1	0.19 +- 0.1
Predicted Cushing's	0.02 +- 0.06	0.98 +- 0.06
Sensitivity	0.84	
Accuracy	0.90	

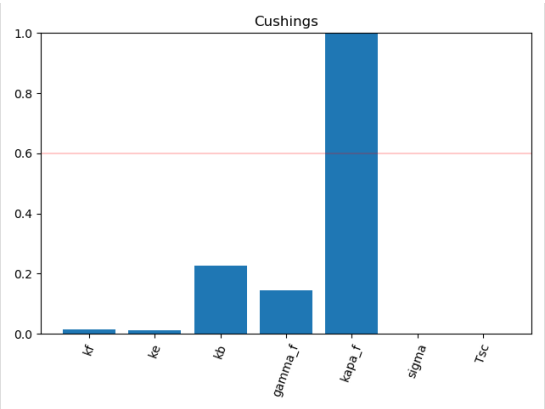
LR Cushing's	True Control	True Cushing's
Predicted Control	0.77 +- 0.17	0.23 +- 0.17
Predicted Cushing's	0.01 +- 0.04	0.99 +- 0.04
Sensitivity	0.81	
Accuracy	0.88	

SVM PrimAldo	True Control	True PrimAldo
Predicted Control	0.58 +- 0.1	0.42 +- 0.1
Predicted PrimAldo	0.25 +- 0.15	0.75 +- 0.15
Sensitivity	0.64	
Accuracy	0.67	

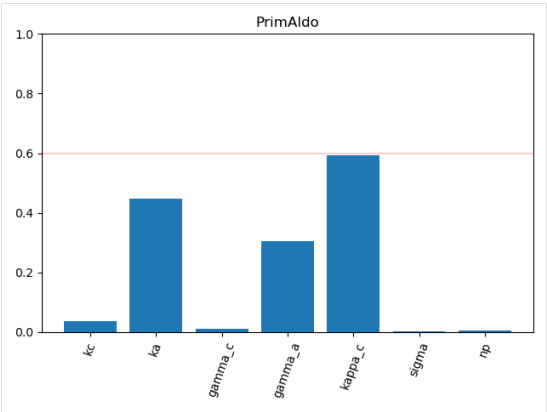
LR PrimAldo	True Control	True PrimAldo
Predicted Control	0.43 +- 0.18	0.57 +- 0.18
Predicted PrimAldo	0.24 +- 0.17	0.76 +- 0.17
Sensitivity	0.57	
Accuracy	0.60	



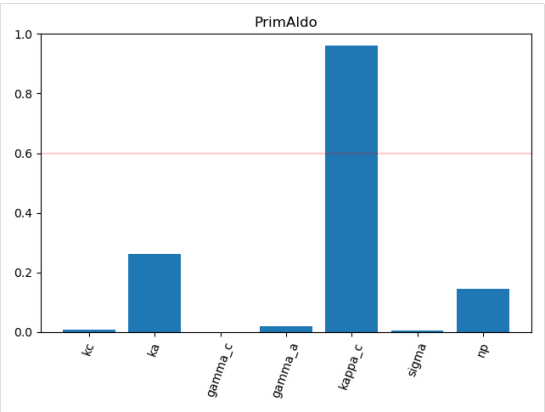
(a) svm model Cushing's partial



(b) lr model Cushing's partial



(c) svm model PrimAldo partial



(d) lr model PrimAldo partial

Figure B.6: Importance plots in the model space with partial adrenal steroid pathway model using threshold 0.6