The prediction of Creatinine and Bilirubin using Machine Learning Methods

JIUXIN WEI

The prediction of Creatinine and Bilirubin using Machine Learning Methods

Jiuxin Wei

A Thesis presented for the degree of
Master of Science (By research)

Department of Computer Science
Durham University
United Kingdom
June 2025

Abstract

This study embarks on an explorative journey into the realm of predictive healthcare, leveraging machine learning (ML) techniques to forecast changes in critical biomarkers - Creatinine and Bilirubin - using Electronic Health Records (EHR). The research employs a comprehensive suite of both supervised and unsupervised learning models, including Gradient Boosting Regressor (GBR), Extra Trees Regressor (ETR), Multilayer Perceptron (MLP), alongside oversampling variants denoted as GBR_o, ETR_o, and MLP_o, and a weight-adjusted MLP model (MLP_w). Additionally, unsupervised approaches such as One-Class Support Vector Machines (SVM), K-Means Clustering, and the Local Outlier Factor (LOF) model are applied to delineate anomalies within the data, presenting a holistic approach to data analysis.

This thesis critically assesses the effectiveness of these models in handling the inherent imbalances and complexities within EHR data, particularly focusing on the predictive accuracy for Creatinine and Bilirubin levels. Oversampling techniques are meticulously applied to rectify class imbalances, enhancing the models' sensitivity towards less prevalent, yet clinically significant outcomes. The comparative analysis highlights the nuanced interplay between model choice, data preprocessing techniques, and the specific characteristics of the biomarkers in question, providing insightful implications for clinical applications. On evaluation with the 825 patients' data, the model achieved sensitivities of 95% (23/24) in the data labelled change of Creatinine, 79% (635/801) in not change of Creatinine,70% (87/124) in the data labelled change of Bilirubin, and 72% (509/701) in not change of Bilirubin.

The findings reveal a varied performance landscape across models and biomarkers, underscoring the importance of tailored approaches in predictive healthcare modeling. Supervised models demonstrated commendable accuracy in majority scenarios, while oversampling techniques offered nuanced benefits, particularly in bolstering the models' ability to detect significant changes in biomarker levels. The study further illuminates the challenges associated with EHR data, including variability, dimensionality, and quality issues, proposing avenues for future research focused on advanced preprocessing techniques, feature selection, and the exploration of deep learning models to surmount these obstacles.

In essence, this research contributes to the burgeoning field of medical informatics by showcasing the potential of ML models to advance predictive diagnostics and personalized medicine, ultimately aiming to enhance patient care through early detection and monitoring of health indicators.

Declaration

The work in this thesis is based on research carried out at the Department of Computer Science, Durham University, United Kingdom. No part of this thesis has been submitted elsewhere for any other degree or qualification and it is all my own work unless referenced to the contrary in the text.

Acknowledgements

First and foremost, I would like to thanks my supervisor, Dr. Noura Al Moubayed, for her constant patience and guidance throughout my studies. She provides me the opportunity to carry out this research. At times when I found myself at a crossroads, lacking clarity and direction regarding my research topic, Dr. Al Moubayed always inspire me and give me some good ideas to pursue. Without her kindly help, I could not finish my research study.

Additionally, I would like to express my gratitude to her Ph.D, students who helped me during my study year. Their collaborative spirit and expertise contributed significantly to my academic development.

Lastly, but most importantly, I owe a debt of gratitude to my parents. Despite facing various challenges as a family, their unwavering support and presence have been a constant source of strength and encouragement for me.

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

1.1 Creatinine and Bilirubin

Creatinine is a byproduct of creatine and creatine phosphate breakdown in muscle and protein metabolism, which is continuously released by the body [1, 4]. The process of creatinine production is irreversible and proceeds at a constant rate (i.e., about 1.1% of the body creatine and 2.6% of creatine phosphate per day is converted to creatinine) (Fig. 1.1) [5]. Serum creatinine concentration is widely interpreted as a measure of glomerular filtration rate (GFR) and is used as an index of kidney function. [6]. Reliable serum creatinine measurements in estimating GFR are critical to global public health efforts to improve chronic kidney disease (CKD) diagnosis and treatment. [7]. Serum Creatinine is not only regarded as a potentially dangerous sign of chronic kidney disease (CKD) but also one of the main criteria for defining acute kidney injury (AKI). The severity of AKI depends on the magnitude of the increase in serum creatinine levels or the decrease in urine output. [8]. Furthermore, creatinine could be used to calculate the sarcopenia index, which can be used to estimate muscle mass [9]. In conclusion, creatinine is useful for the assessment of kidney function and muscle mass.

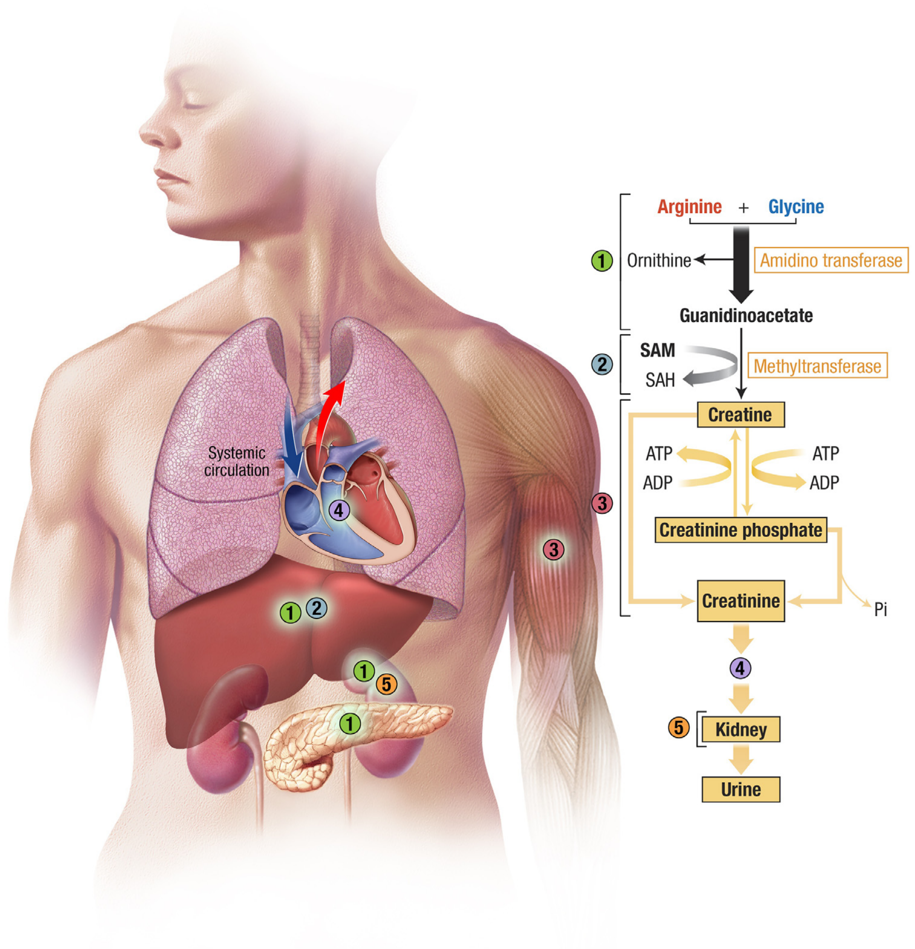

Figure 1.1: Creatine metabolism pathway showing synthesis from arginine and glycine, phosphorylation to creatine phosphate, and excretion as creatinine through the kidneys. Adapted from [1].

Bilirubin is the end product of heme catabolism in mammals, which is normally considered as a waste product that must be excreted [10]. Bilirubin is formed by the cleavage of haemoglobin as shown in Figure 1.2 [11]. Serum bilirubin could be a protective marker for Nonalcoholic fatty liver disease (NAFLD), as the levels of bilirubin are inversely associated with the prevalence of NAFLD [2]. Bilirubin is not only used to estimate liver diseases but also a powerful signalling molecule. For example, Bilirubin has a protective effect on autoimmune and inflammatory diseases because it inhibits almost all immune system effectors [12]. It has also been identified as a cardio and metabolic protective factor with therapeutic implications [13].

Both Bilirubin and Creatinine levels in the blood can be influenced by a wide array of factors including age, gender, muscle mass, diet, and medication, which

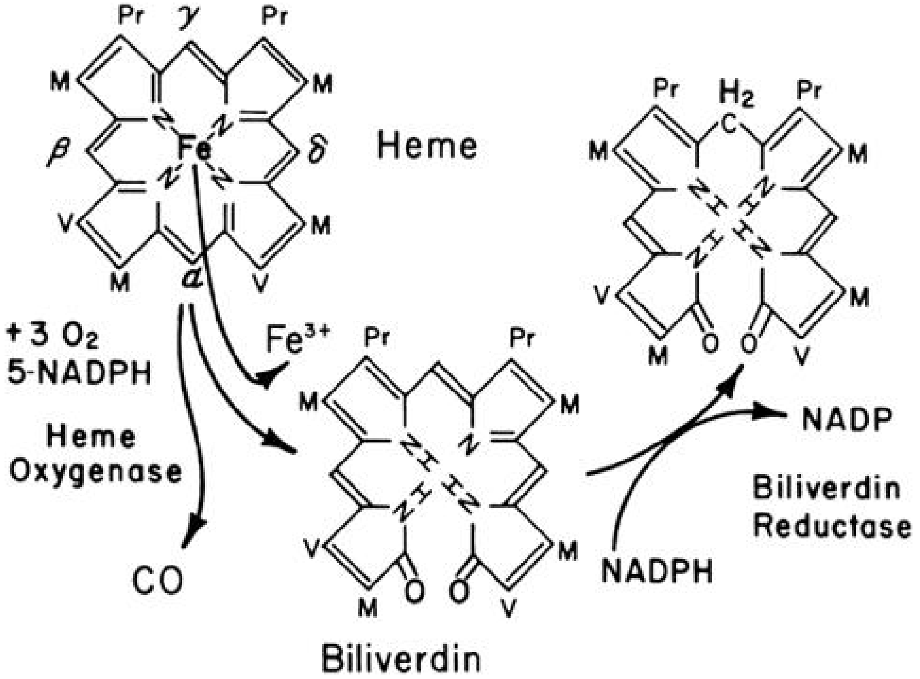

Figure 1.2: Heme catabolism pathway illustrating the conversion of heme to biliverdin via heme oxygenase, followed by reduction to bilirubin. Adapted from [2].

introduces significant variability. This variability can make it difficult to develop models that accurately predict levels across diverse populations [4]. However, accurate predictions of Bilirubin and Creatinine levels can facilitate early detection and monitoring of kidney and liver diseases. Predictive models can certainly contribute to personalized medicine by identifying individuals at risk of significant changes in Bilirubin and Creatinine levels before clinical symptoms manifest. This allows for tailored treatment plans that consider the individual's risk profile and disease trajectory. Early intervention based on these predictions can lead to improved patient outcomes and reduced healthcare costs [14].

1.2 EHR data

The prediction of Bilirubin and Creatinine levels is based on the electronic health records (EHR). The development of information technology has changed the way healthcare is performed and documented. Currently, databases in hospitals automatically capture structured data relating to all aspects of care, including laboratory test results, diagnoses, physicians' notes, and treatments. EHRs generally contain demographic, vital statistics, administrative, claims (medical and pharmacy), clinical, and patient-centred (e.g., originating from health-related quality-of-life instruments, home-monitoring devices, and frailty or caregiver assessments) data [15].

This detailed information is crucial for predicting health outcomes, such as changes in Bilirubin and Creatinine levels, by identifying patterns and correlations within the data. Furthermore, by leveraging EHR data, predictive models can analyse real-time updates to a patient's health records, enabling early detection of potential health issues. For example, an upward trend in Creatinine levels over time could indicate deteriorating kidney function, prompting early intervention before the condition progresses further. Also, the detailed patient data contained within EHRs allow for the development of personalized predictive models. Such models can consider individual patient characteristics, such as age, sex, underlying health conditions, and genetic information, to make more accurate predictions regarding Bilirubin and Creatinine levels for each patient, thereby facilitating personalized care plans.

In summary, EHR data play a pivotal role in the development and implementation of predictive models for Bilirubin and Creatinine levels, enhancing the ability to monitor, detect, and manage potential health issues effectively. The integration of predictive analytics with EHR systems represents a significant advancement in the pursuit of proactive and personalized healthcare.

1.3 Machine Learning

In the past two decades, machine learning has progressed dramatically. Machine learning is the scientific study of how computers learn from data, which is seen as a subset of artificial intelligence. Like many algorithms, machine learning algorithms must be designed precisely and updated iteratively to be effective. Machine Learning includes various types, with supervised and unsupervised learning being most relevant to biomedical prediction tasks. Both supervised and unsupervised learning construct a mathematical model to solve an optimisation solution problem. The data for supervised learning contains labels, which is the learning goal of supervised learning. The fundamental goal of machine learning is the ability to generalise, expecting to learn rules for sample partitioning based on the labelled features of the training data, and to apply this rule to unknown data, thus completing predictions on unknown data. Unsupervised learning does not use labelled data; instead, it learns the intrinsic relationships and structural information of the samples, which in this thesis is used for anomaly detection to identify unusual patterns in Creatinine and Bilirubin levels.

The use of machine learning-based methods for predicting Bilirubin and Creatinine levels, as well as other health-related predictions using EHR data, offers several compelling advantages over traditional statistical approaches. Human health and diseases are influenced by a complex interplay of genetic, environmental, lifestyle, and socio-economic factors. Machine learning models excel at capturing these complex, non-linear relationships within large datasets, such as those found in EHRs, which traditional statistical models may not handle as effectively. Moreover, machine learning algorithms can learn from a wide array of data points for each individual, including past medical history, lab results, and treatment responses. This enables the development of personalized health predictions and treatments tailored to the unique characteristics of each patient. With the ever-growing volume of EHR data, machine learning models can efficiently process and analyse large datasets to identify patterns and insights quickly. This scalability is crucial for applying predictive models across large populations [16]. In addition, the employment of machine learning with EHR data have shown great success [17]. Nonetheless, there are very limited studies about predicting the levels of Creatinine and Bilirubin using EHR data.

1.4 Thesis Contributions and Structure

In the current study, we employed a variety of machine learning algorithms to analyze EHR data with the objective of predicting levels of Creatinine and Bilirubin. This investigation encompasses both supervised and unsupervised learning paradigms. Specifically, we utilised Gradient Boosting Regressor (GBR), Extra Trees Regressor (ETR), and Multilayer Perceptron (MLP) as our supervised learning models. In the realm of unsupervised learning, we explored the application of One-Class Support Vector Machines (SVM), Local Outlier Factor (LOF), and K-Means Clustering techniques. Additionally, to address potential issues of data scarcity and imbalanced datasets, we implemented several oversampling techniques, notably the Random Oversampler, to augment the EHR dataset. This approach aims to enhance the predictive accuracy of our models. In summary, the primary contributions of this manuscript include:

(1) Applying both supervised and unsupervised learning models to predict the levels of Creatinine and Bilirubin.

(2) Comparing the machine learning models used in this paper and finding the one that works best.

(3) The predicted values are classified according to a medical formula into two categories of great and not great change to help doctors diagnose.

This thesis presents a number of machine learning models including supervised models and unsupervised models. Chapter 2 thoroughly reviews the literature about involving EHR data and machine learning method in prediction, and the prediction of Creatinine and Bilirubin, specifically. Chapter 3 introduces the methods we used in this thesis including the oversampling method which is used to solve the data imbalance problem, three supervised learning models, and three unsupervised learning models to predict the levels of Creatinine and Bilirubin. Chapter 4 describes how the experiments was conducted. Chapter 5 and Chapter 6 summarise and discuss the research questions, process, and results.

1.4.1 Research Questions

This thesis investigates and applies some machine learning models to predict the levels of Creatinine and Bilirubin. To achieve the research goals, we conclude the following research questions.

1. How accurately can machine learning models predict Creatinine and Bilirubin levels using EHR data, and which features are most predictive?

2. Which machine learning model achieves the best performance in our experiment?

3. Can the predicted Creatinine and Bilirubin levels be effectively classified into clinically meaningful categories to assist healthcare professionals in diagnosis and treatment decisions?

CHAPTER 2

Related Work

This chapter reviews the related work in three main parts. The first part is the introduction and application of EHR data in medicine. The second part reviews the prediction of Machine Learning in various industries, especially in Medicine. The final part presents the literature on the prediction of levels of Creatinine and Bilirubin.

2.1 EHR Data

An electronic health record (EHR) is known as the collection of patient and population stored health data in a digital format [18]. The amount of digital information stored in electronic health records has increased dramatically over the last decade. Although these records were designed primarily for archiving patient information and performing administrative health care tasks, many researchers have discovered secondary uses for them in a variety of clinical informatics applications [19]. In recent years, there is a large volume of published studies describing the use of EHR data in medical decision support tasks [20]. Several studies have developed prediction and detection models based on EHR data. For example, Nitzan et al. [21]

proposed a model to predict Gestational diabetes mellitus (GDM) with nationwide EHR data. Di et al. [22] combined PubMed knowledge and EHR to construct a model for pancreatic cancer prediction. Matthew and Noura [23] established a model for the accurate detection of adversarial samples on EHR and chest X-ray (CXR) data. Michael et al. [24] utilized EHR data to detect and classify type 1 versus type 2 diabetes.

EHR data often come from various sources, leading to significant heterogeneity in terms of format, structure, and coding systems. Additionally, data quality issues such as missing values, errors, and inconsistencies further complicate data preprocessing and analysis [25]. This results in imbalanced datasets where the number of instances in one class significantly outweighs the other, complicating model training and skewing performance metrics. He et al. [26] provided a comprehensive overview of methods for learning from imbalanced data, including resampling techniques and specialized algorithms designed to improve model performance on minority classes. EHR data can be extremely high-dimensional, with numerous features collected from patient records. This dimensionality poses computational challenges and increases the risk of overfitting, requiring sophisticated feature selection and dimensionality reduction techniques to build effective predictive models. Bellazzi et al. [27] explored the application of machine learning techniques in clinical medicine, including methods for handling the high dimensionality of EHR data, such as feature selection and dimensionality reduction.

2.2 Machine Learning in prediction

In recent years, with the development of big data and data science, machine learning has been successfully applied to a range of industries, such as consumer services, fault diagnosis in complex systems, and logistics chain control. [28] With a focus on clinical problems, machine learning is based on a large number of different types of clinical data, which is processed accordingly with the help of tools such as statistical analysis and bioinformatics, to provide more help to doctors and patients. Machine learning has many applications in medicine. For instance, a number of studies have used Machine Learning techniques to predict the risk assessment and survival of cancer. [29]. Various machine learning techniques have been applied in medical imaging across the decades. [30] Senthilkumar M et al. [31]used several machine learning models to improve the accuracy in the prediction of cardiovascular disease. Furthermore, supervised learning is often used to estimate risk. [16] Gradient Boosting Regressor (GBR) is a supervised learning model, which is intensively used in prediction. Yanru Zhang et al. [32] have demonstrated that the GBR model has significant advantages in predicting freeway travel times. Simon N et al. [33] have indicated that the GBR model could perform well in the prediction of major chronic diseases.

Deep learning is seen as a part of machine learning methods. Deep learning is based on neural networks that learn the intrinsic logic and features embedded in sample data to achieve a specific task. With the proliferation of high-performance Graphic Processing Units (GPUs), the average cost of computational power is becoming lower and lower, laying the foundation for research into deep learning models that rely on large amounts of computation. Deep learning is one of the mainstream research directions in artificial intelligence and has been applied to a wide range of fields now. [34] Deep learning is a powerful technique for predicting protein structure in medicine. [35] Deep learning networks that are commonly used include Deep Neural Networks(DNN), Convolutional Neural Networks(CNN), Recurrent Neural Networks(RNN), and Generative Adversarial Networks(GAN). Deep learning is a powerful technique for predicting protein structure. Multilayer perceptron (MLP) is a popular deep learning model, which is a type of fully connected feedforward artificial neural network (ANN). Many researchers have applied the MLP model to improve medicine. Dimitrios H et al. [36] utilized the MLP and probabilistic neural networks (PNNs) models for Osteoporosis risk prediction. Mohamed et al. [37] used MLP and Long Short Term Memory (LSTM) techniques to predict heart disease.

2.3 The prediction of Creatinine and Bilirubin

Creatinine and Bilirubin are two common and clinically significant laboratory values in medical records. There is a large volume of published studies using the value of them to help doctors treat diseases. Chicco D and Jurman G [38] demonstrated that serum creatinine and ejection fraction are the two most relevant factors in predicting the survival of patients with heart failure. Song X et al. [39] used creatinine and other medical values to predict acute kidney injury with machine learning and logistic regression models. Inoguchi T et al. [40] used a gradient boosting decision tree (GBDT) model and a Cox proportional hazard regression model to assess the association between serum bilirubin levels and cancer risk, and they demonstrated that serum bilirubin may have a protective effect against certain types of cancer. Akter S et al. [41] developed a method to assess the natural progression of liver disease by evaluating the serum total bilirubin (TB) and seven other biochemical parameters. However, there are very few studies predicting the value of Creatinine and Bilirubin using EHR data.

A number of studies in the field of medicine have been carried out to predict the levels of Creatinine and Bilirubin in the past. For example, Cockcroft D et al. [42] developed a formula to predict creatinine clearance (Ccr) from serum creatinine (Scr). However, there are very few studies predicting them using artificial Intelligence technologies. Dauvin A et al. [43] utilized several machine learning algorithms to predict pre-admission creatinine and baseline hemoglobin intensive care patients with the MIMIC-III database. They used gradient-boosted trees, random forest, and logistic regression models. Wang W et al. [44] adopted ensemble learning techniques to predict creatinine value from 23 features, and then combined the predicted creatinine value with the original 23 features to assess the risk of Chronic Kidney Disease (CKD). Surachate et al. [45] proposed a supervised machine learning model to distinguish among low, moderate, and high normal serum creatinine by evaluating tear creatinine. Ghosh E et al. [46] demonstrated that the baseline serum creatinine level could be estimated better using a gradient boosting model than the back-calculated estimated serum creatinine level. In my study, I used different models and a different dataset from the above studies. I applied gradient-boosted trees, extra trees and multilayer perceptron models on the dataset and compared them to find the best performance. For the research on predicting Bilirubin, there is no study predicting the value using EHR data. Aune A et al. [47] designed a smartphone-based tool to estimate bilirubin levels from digital images. They combined colour analysis of digital images with physics-based modelling of light transport in skin to predict bilirubin levels in newborn infants. Imant etal. [48] trained an ensemble, that combines a logistic regression with a random forest classifier, to enhance the early prediction of clinically relevant neonatal hyperbilirubinemia with the serial measurements of total serum bilirubin in the first two weeks of life.

CHAPTER 3

Methods

In this chapter, I provide a brief overview of the regression models employed to address the objectives of this thesis. This chapter proceeds by first discussing the supervised learning models used in this thesis, and some unsupervised learning models followed.

3.1 Oversampling

Oversampling is a technique employed to ameliorate the imbalance within a training dataset by augmenting the number of instances in under-represented classes. This approach is particularly beneficial in datasets where the disparity between class distributions is significant. In our study, we observed a pronounced imbalance, with the quantity of samples exhibiting no significant change vastly outnumbering those with considerable change. To address this imbalance, we implemented four distinct oversampling strategies: Random Oversampler, Synthetic Minority Over-sampling Technique (SMOTE), Adaptive Synthetic (ADASYN) Sampling, and Borderline SMOTE Oversampler.

The Random Oversampler method enhances the minority class's presence by duplicating existing samples randomly. Mathematically, this can be expressed as:

$$D' = D \cup \left\{ x_j^{(copy)} | x_j \in D_{minor}, \text{ randomly selected with replacement, } N \text{ times} \right\} \tag{3.1}$$

where D' is the augmented dataset, D is the original dataset, D_{minor} represents the minority class samples, and N is the number of duplication.

The SMOTE algorithm generates synthetic samples by interpolating between existing minority class samples [49]. The interpolation process for a sample x_i can be described by:

$$x_{new} = x_i + \lambda(x_z - x_i) \tag{3.2}$$

where x_z is a randomly chosen sample from the k nearest neighbours of x_i in the minority class, and λ is a random number between 0 and 1.

ADASYN extends SMOTE by adjusting the number of synthetic samples generated for each minority class sample based on its "difficulty" in being correctly classified, measured by its k-Nearest neighbours [50]. The generation of synthetic samples is guided by:

$$x_{new} = x_i + \lambda(x_{nn} - x_i) \tag{3.3}$$

where x_{nn} is a neighbour of x_i selected based on the adaptive density distribution.

Borderline SMOTE focuses on samples that are near the decision boundary (the border) and might be misclassified [51]. Synthetic samples are created by interpolating between these borderline samples and their nearest neighbours in the minority class that are also near the decision boundary.

In our experimental framework, these four oversampling methods were deployed to enrich the dataset with more "great change" instances before applying regression models for predictive analysis. The performance of these methods was evaluated based on the efficacy of subsequent predictive modelling.

3.2 Gradient Boosting

Gradient boosting is a machine learning technique that is commonly used in regression and classification tasks. It gives a prediction model in the form of an ensemble of weak prediction models, usually decision trees [52]. When a decision tree is used as the weak learner, the resulting algorithm is known as gradient-boosted decision trees (GBDT), and it typically outperforms random forest [53]. Gradient boosting of regression trees (GBR) produces competitive, highly robust, interpretable procedures for both regression and classification [54]. Its prediction model can be represented as:

$$\hat{y}_i = \sum_{k=1}^{k} f_k\left(x_i\right) \tag{3.4}$$

where k is the total number of trees, f_k is the tree numbered k, y_i is the prediction result of the sample x_i.

The prediction model follows the forward distribution addition method, which generates a new regression tree at each iteration, and the new tree will keep fitting the residuals of the previous tree to continuously improve the previous experimental results, as shown in the equation below:

$$\hat{y}_i(t) = \sum_{k=1}^{t} f_t\left(x_i\right) = \hat{y}_i(t-1) + f_t\left(x_i\right) \tag{3.5}$$

In the formula, t is the combined t-tree; $y_i(t)$ is the prediction result of the combined t-tree model for sample x_i, $y_i(t-1)$ is the prediction result of the combined $(t-1)$-tree model for sample x_i, $f_t(x_i)$ is the estimated value of the tree numbered t model for the current round of losses.

For each iteration of GBDT, the loss function in the current mode is used to negative gradient of the loss function under the current model to fit the estimate of the current round's loss (i.e., the residual estimate value). This allows the loss function to be reduced as quickly as possible in each training round, and converges to the residual estimate as quickly as possible. In this way, the loss function can be reduced as fast as possible in each training round, and converge to the local optimal solution or the global optimal solution as soon as possible. The negative gradient of the loss function for the ith sample of round t is expressed as:

$$r_{ti} = -\left|\frac{l\left(y_i, \hat{y}_i\right)}{\hat{y}_i}\right|_{f(x)=f_{i-1}(x)} \tag{3.6}$$

The best fit on each leaf node R_{tj} that minimizes the loss function The values c_{tj} are summed to obtain the estimate of the tree numbered t model for the current round of losses $f_t(x_i)$

$$f_i\left(x_i\right) = \sum_{i=I}^{J} c_{tj} I\left(x_i \in R_{tj}\right) \tag{3.7}$$

The GBR model parameters mainly include the estimators (the number of boosting stages to perform), learning rate and loss function.

In summary, gradient boosting models are often synonymous with high predictive performance, especially in tabular data challenges. They are capable of handling mixed types of data: numerical and categorical. However, the training of the algorithm can be time-consuming, as trees are built sequentially. As the number of trees increases, the model becomes more complex and harder to interpret compared to simpler models.

3.3 Multilayer perceptron

The Multilayer Perceptron (MLP) is a prevalent model in the realm of neural networks, operating under the principles of supervised learning. At its core, the MLP comprises neurons, which are the fundamental processing units. Each neuron receives inputs, xx_1, x_2,..., x_n , and produces an output, y, as depicted in Fig.3.2, a representation of an MLP network.

The inputs to a neuron are weighed by a set of coefficients, w_1, w_2,..., w_n, reflecting the relative importance of each input to the neuron's operation. These weights modulate the inputs, which are then aggregated and subjected to a bias term, b, to compute the neuron's net input, μ. This process can be formalized by the weighted summation formula:

$$\mu = \sum_{i=1}^{n}(w_i \cdot x_i) + b \tag{3.8}$$

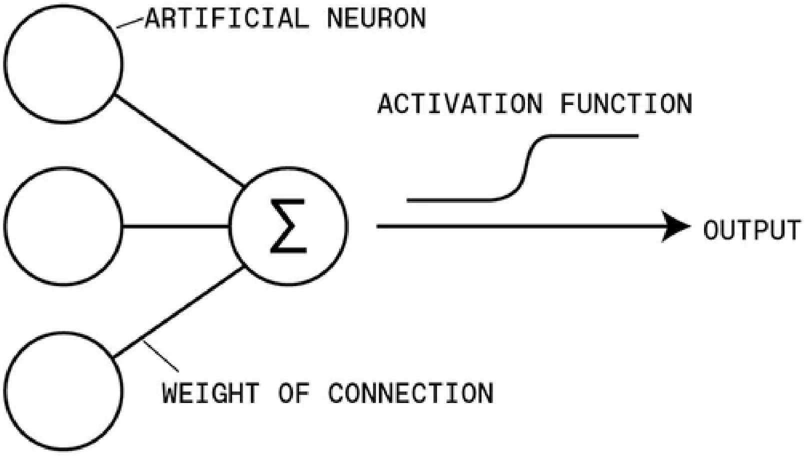

Figure 3.1: Structure of a single artificial neuron showing inputs $(x_1, x_2, ..., x_n)$, weights $(w_1, w_2, ..., w_n)$, summation function (Σ), activation function, and output. This diagram illustrates the basic computational unit that forms the building blocks of neural networks [3].

Alternatively, considering the bias b as an additional weight w_0 paired with a constant input of 1, the net input computation becomes:

$$\mu = \sum_{i=0}^{n} (w_i \cdot x_i) \tag{3.9}$$

Subsequently, an activation function, f , is applied to μ to generate the neuron's output: $y = f(\mu)$, which introduces non-linearity into the model, enabling it to capture complex relationships in the data.

Throughout the training phase, the MLP is fed with input-output pairs, iteratively adjusting its weights to minimize the discrepancy between the predicted outputs and the actual targets. This process continues until the network achieves an optimal representation of the underlying data distribution, at which point it can be used for making predictions on new, unseen data. MLPs can model complex non-linear relationships between inputs and outputs, making them suitable for a wide range of tasks, from regression to classification.

Theoretically, MLPs with at least one hidden layer and appropriate activation functions can approximate any continuous function to any desired degree of accuracy, a principle known as the Universal Approximation Theorem. This theorem states that a feedforward network with a single hidden layer can approximate any

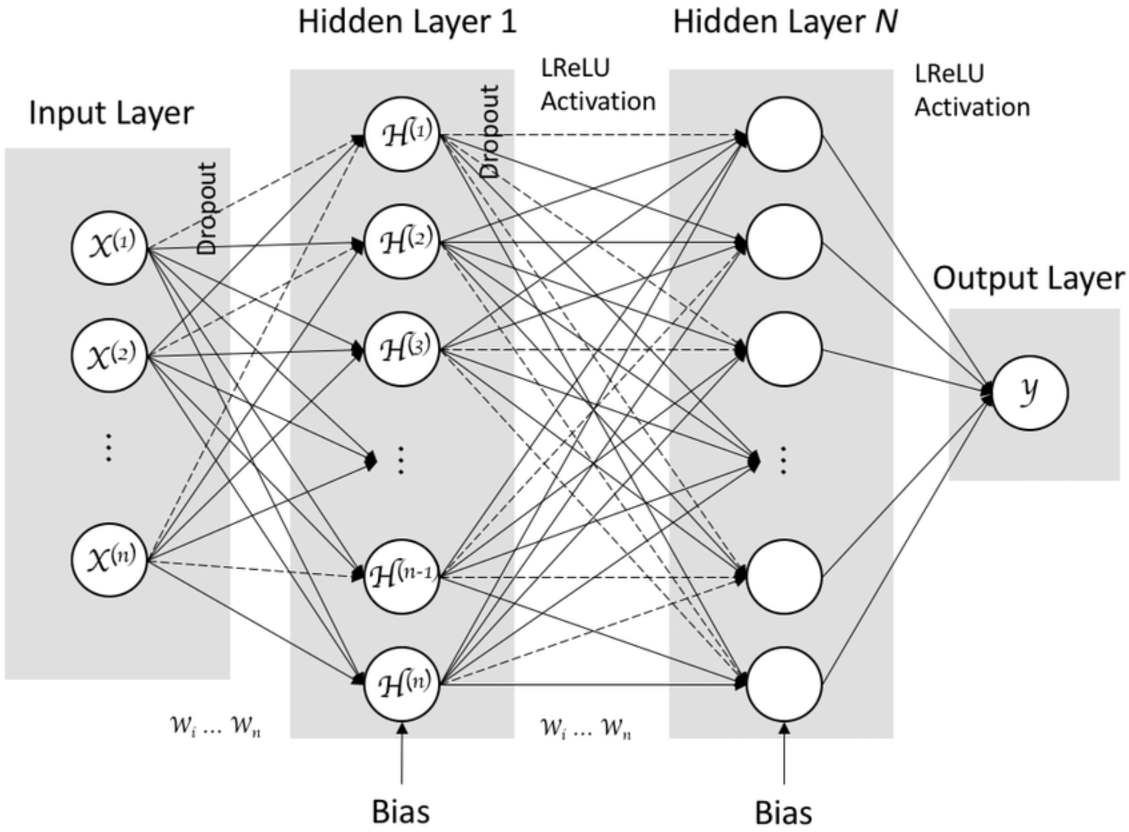

Figure 3.2: **A simple artificial neuron model.** This diagram illustrates the structure of an artificial neuron, a fundamental building block in multilayer perceptrons and deep learning architectures. Each input (depicted as circles) is assigned a weight, and the weighted sum of these inputs is computed. This sum is then passed through an activation function, which introduces non-linearity to the model. The resulting output is used either as the final prediction or as input to subsequent layers.

continuous function on a compact subset of $\mathbb{R}^n$ to arbitrary accuracy, provided the activation function is non-constant, bounded, and monotonically-increasing. However, while the theorem guarantees the existence of such an approximation, it does not specify the required number of neurons or guarantee that standard training procedures will find the optimal parameters.

Without proper regularisation, MLPs can overfit the training data, learning noise rather than the underlying pattern, which degrades their performance on unseen data. Training an MLP, especially with large datasets or architectures, can be computationally demanding and time-consuming. The performance of MLPs is highly sensitive to the choice of architecture (e.g., the number of layers and neurons) and hyperparameters (e.g., learning rate), requiring extensive tuning for optimal results.

3.4 Extra Trees

Extreme Randomized Trees (ERT), also known as Extra-Trees, represent an innovative approach within tree-based ensemble methods, distinguishing themselves from their counterparts such as Random Forests and Deep Forests [55]. The foundational principle of ERT is to employ decision trees as base estimators, utilizing the entire dataset for training each tree and incorporating randomness in the selection of features and the split decisions. This methodology is articulated in the seminal work by Geurts et al [56].

In ERT, the algorithm constructs multiple decision trees, with each tree trained on the complete dataset. Unlike Random Forests, which employ bootstrapping to generate varied training subsets, ERT leverages the entire data for every tree, aiming to maximize the utilization of available information. The key aspect of ERT is the random selection of cut-points for each feature, defined mathematically as:

$$\theta_{feature,split} \sim U(D_{feature}) \tag{3.10}$$

where $\theta_{feature,split}$ denotes the randomly chosen threshold for a given feature from its distribution U over the dataset D. The optimal split is determined through an evaluation of these random thresholds, ensuring a diverse set of decision rules across the ensemble.

The final prediction of the ERT model is obtained by averaging the predictions from all individual trees:

$$y_{pred} = \frac{1}{N} \sum_{i=1}^{N} T_i(x) \tag{3.11}$$

where N is the number of trees, $T_i(x)$ represents the prediction of the i th tree for input x, and y_{pred} is the ensemble's output.

By averaging over numerous trees, ERT effectively mitigates variance, enhancing stability and robustness. Training each tree on the full dataset ensures that all available information is leveraged, potentially improving predictive performance.

The introduction of randomness in feature selection and splits can also reduce model bias, making ERT less prone to overfitting compared to more deterministic tree models. However, training on the entire dataset and evaluating numerous random splits can be computationally demanding, especially with large datasets. The ensemble nature and the random selection process in ERT compromise interpretability, making it challenging to extract intuitive rules from the model. While randomness can introduce diversity, it may also lead to splits on noisy features, potentially affecting the model's generalization capability.

3.5 Unsupervised learning models

We used three unsupervised learning models to classify the Exceptional values of Creatinine and Bilirubin: one-class suport vector machine (SVM), Local Outlier Factor (LOF), K-Means Clustering.

3.5.1 One-Class SVM

The One-Class Support Vector Machine (SVM) is an extension of the traditional SVM framework, tailored specifically for anomaly detection and novelty detection tasks. Unlike the conventional SVM that distinguishes between two or more classes, the One-Class SVM focuses on identifying data points that deviate from the norm, effectively distinguishing between normal instances and outliers within a dataset. The One-Class SVM aims to find a function $f(x)$ that captures the region in the feature space populated by the majority of data points. This function is designed to return a positive value for regions with dense data (normal instances) and a negative value elsewhere (anomalies). The decision function for a given input x can be represented as:

$$f(x) = sign(\langle w, \phi(x) \rangle - \varrho) \tag{3.12}$$

Here, $\phi(x)$ denotes a mapping of the input vectors x into a higher-dimensional feature space, w is the weight vector orthogonal to the hyperplane, and ϱ represents the offset of the hyperplane from the origin in the feature space.

The objective of the One-Class SVM is to maximize the margin around the hyperplane subject to most data points lying on the side of the hyperplane corresponding to the target class. This leads to the optimization problem:

$$\min_{w,\xi,\varrho} \frac{1}{2} \|w\|^2 - \varrho + \frac{1}{vn} \sum_{i=1}^{n} \xi_i \tag{3.13}$$

subject to the constrains:

$$\langle w, \phi(x) \rangle \geq \varrho - \xi_i, \xi_i \geq 0, i = 1, 2, ..., n \tag{3.14}$$

where ξ_i are slack variables allowing for a fraction of data points to lie on the opposite side of the hyperplane, and v is a parameter that controls the trade-off between the fraction of outliers and the decision function's margin.

One-Class SVM is highly effective in scenarios where the goal is to identify data points that significantly deviate from the majority of the dataset. The formulation as a quadratic optimization problem ensures that the solution is a global optimum, providing consistency in model performance. Through the kernel trick, One-Class SVM can operate in a high-dimensional feature space, enabling it to capture complex patterns in the data. The commonly used kernel functions include Linear Kernel (LK), Polynomial Kernel (PK) and Radial Basis Function (RBF). In this paper, RBF is used as the kernel function because of its strong nonlinear mapping capability, which can be written as following:

$$K\left(x_i, x_j\right) = \exp\left(-\delta \cdot \|x_i - x_j\|^2\right) \tag{3.15}$$

where δ is the parameter of the kernel function representing the spatial extent that a particular training sample can reach. The performance of the One-Class SVM is sensitive to the choice of kernel and its parameters (e.g., δ in the RBF kernel) and the value of v, which can make the tuning process challenging. Similar to traditional SVMs, the computational complexity of One-Class SVM can become a concern with very large datasets, particularly due to the need to invert a matrix in the optimization step.

3.5.2 Local Outlier Factor

The Local Outlier Factor (LOF) algorithm is an unsupervised method used to identify outliers in a dataset. It operates on the premise that the density around a normal instance is similar to the density around its neighbours, whereas outliers lie in sparser regions. The LOF provides a way to quantify the local deviation of a given data point with respect to its neighbours, allowing for the identification of instances that significantly diverge in density from their surrounding area. For each data point x, LOF computes a score that reflects its degree of outlier-ness based on the local density. The process involves several steps:

(1) Let k be a positive integer and the k-th distance of a data point p be denoted as $d_k(p)$, i.e. the distance from the data points k-th away from data point p to data point q. The distance between a data points q and p in the data set C is denoted as $dist_k(p, q)$, and there are two cases that could make $d_K(p) = dist_k(p, q)$: i. There exist at least k data pointss $q' \in C\{x \neq p\}$ satisfying $d\left(p, q'\right) \leq d(p, q)$, ii. There are at most $k - 1$ data pointss $q' \in C\{x \neq p\}$ satisfying: $d\left(p, q'\right) < d(p, q)$.

(2) The k-th distance domain of a data points p is the set of all data pointss whose distance from p is less than the k-th distance, denoted as:

$$\left|N_{d_k(p)} = \{q \in C \mid d(p, q) \leq d_k(p)\}\right| \geq k \tag{3.16}$$

(3) The reachable distance from data points p to q is denoted as $reach - dist_k(p, q)$. It is at least the k-th distance from p, or the true distance between q and p. It could be represented as:

$$reach - dist_k(p, q) = max\left\{(d_k(p), d(p, q)\right\} \tag{3.17}$$

(4) The local reachable density of a data points p denotes the inverse of the average reachable distance from the data pointss to p in the k-th domain of p, denoted as $lrd_k(p)$,i.e.:

$$lrd_k(p) = \frac{1}{\left[\frac{\sum_{q \in N_k(p)} \text{ reach } - dist_k(p,q)}{|N_k(p)|}\right]} \tag{3.18}$$

, where, $N_k(p)$ is the set of k nearest neighbour of p.

(5) The local outlier factor of a data points p denotes the mean of the ratio of the locally reachable density of the k-th domain $N_k(p)$ of p to the locally reachable density of p, denoted as $LOF_k(p)$, so it could be represented as:

$$LOF_k(p) = \frac{\sum_{q \in N_k(p)} \frac{lrd_k(q)}{lrd_k(p)}}{|N_k(p)|} \tag{3.19}$$

A LOF score significantly hreater than 1 indicates an outlier. Unlike global methods, LOF considers the local density variation, making it effective in datasets with varying densities. LOF can identify outliers that may not be detectable with distance-based or global methods due to its focus on local neighbourhoods. As an unsupervised method, LOF requires no labeled data for training, making it suitable for datasets where outlier labels are not available. However, The choice of k (the number of neighbours) can significantly affect the outcome. Too small or too large values might lead to misleading results. The need to compute distances between all pairs of points makes LOF computationally intensive, especially for large datasets.

3.5.3 K-Means Clustering

The K-Means algorithm is a widely-used method for partitioning a dataset into K distinct, non-overlapping subsets or clusters. It aims to minimize the variance within each cluster, effectively grouping data points based on feature similarity. The algorithm iterates through two main steps: assignment of data points to the nearest cluster centre and update of cluster centres based on the current cluster assignments. Given a dataset $X = \{x_1, x_2, x_3, ..., x_n\}$ consisting of n data points and a pre-specified number of clusters K, the goal of K-Means is to find a set of K cluster centres $C = \{c_1, c_2, c_3, ..., c_K\}$ that minimises the within-cluster sum of squares (WCSS) defined as:

$$WCSS = \sum_{k=1}^{K} \sum_{x \in S_k} \|x - c_k\|^2 \tag{3.20}$$

where S_k represents the set of data points assigned to cluster k with centre c_k. The algorithm proceeds as follows:

(1) Initialization: Select K initial cluster centres, which can be done randomly or by more sophisticated methods like the K-Means++ algorithm.

(2) Assignment Step: Assign each data point to the nearest cluster center:

$$S_k = \{x : \|x - c_k\| \leq \|x - c_j\|, \forall j, 1 \leq j \leq K\} \tag{3.21}$$

(3) Update Step: Update each cluster center to be the mean of the data points assigned to it:

$$c_k = \frac{1}{\|S_k\|} \sum_{x \in S_k} x \tag{3.22}$$

(4) Iteration: Repeat the assignment and update steps until the cluster assignments no longer change or a predefined number of iterations in reached.

K-Means is straightforward to implement and computationally efficient, making it suitable for a wide range of clustering tasks. With optimizations, such as the use of the Elkan algorithm, K-Means can scale to large datasets. However, the choice of initial cluster centres can significantly affect the final outcome, potentially leading to sub-optimal solutions. K-Means assumes clusters to be spherical and evenly sized, which might not be the case in many real-world datasets, leading to poor performance on elongated or imbalanced clusters.

CHAPTER 4

Experiments and Results

This chapter include the experiment conducting procedure and result, it proceeds by: (i)introducing the dataset, and describing how to process the EHR data; (ii) explaining the supervised and unsupervised learning models we used; (iii) Showing the results of our experiments.

4.1 Dataset

The dataset under investigation was meticulously compiled over an extensive period spanning from 2011 to 2018, encompassing medical records from 999 distinct patients. Given the longitudinal nature of this dataset, individual patients often contributed multiple entries, reflecting successive medical evaluations over time.

To increase the dataset size for machine learning purposes, each patient visit was treated as a separate instance rather than aggregating all visits per patient. This transformation converted the original 999 patient records into 4,124 visit-based instances. The process maintained temporal relationships while preserving data integrity for machine learning applications.

This dataset represents a proprietary medical dataset that is not publicly available due to patient privacy regulations and institutional data governance policies. The comprehensive eight-year longitudinal coverage and the novel visit-centric transformation approach constitute significant contributions of this research to the field of medical data preprocessing.

The process maintained temporal relationships while significantly reducing missing data through the visit-based transformation approach.

The dataset's 30 attributes can be bifurcated into two primary categories: 28 features and 2 predictive targets. The features themselves are further categorized into three sub-groups, namely personal information, blood test results, and documented medical conditions, providing a comprehensive overview of each patient's health status.

(1) Personal Information: This sub-category includes demographic and physiological parameters such as ethnicity, sex, prescribed medical regimen, height, weight, and age, offering a foundational understanding of the patient's profile.

(2) Blood Test Results: Encompassing a broad spectrum of hematological assessments, this section includes markers such as Granulocyte-Colony Stimulating Factor (G-CSF), Direct Reactivity (DR), Absolute Neutrophil Count (ANC), Platelet Count Test (PLTs), Hemoglobin levels, Creatinine, Alanine Transaminase (ALT), Bilirubin, Body Surface Area (BSA), and Bolus Drop, providing critical insights into the patient's physiological and metabolic status.

(3) Medical Conditions: This subset documents a range of diagnosed conditions including Diabetes, Cardiovascular diseases, Malignant Hyperthermia (MH), Thyroid disorders, Chronic Ulcerative Colitis (UC), Omeprazole therapy, respiratory rate (Resp), Arthritis, Autoimmune diseases, Epilepsy, Hepatitis B (hepb), and Performance Status (PS), thereby offering a detailed medical history relevant to each patient's health trajectory.

The predictive targets within the dataset are quantified levels of Creatinine and Bilirubin for subsequent patient visits. These biomarkers serve as critical indicators of renal and hepatic function, respectively, and their predictive modelling is of paramount importance for proactive healthcare management and intervention planning.

4.1.1 Data-Preprocessing

For the dataset, an additional categorical column was incorporated to delineate between instances exhibiting 'great change' and 'not great change' in the Creatinine and Bilirubin values across two consecutive assessments. This classification facilitates a more intuitive evaluation of potential significant alterations in the patient's blood biochemistry. To accommodate the analytical framework, all categorical variables were transformed into dummy (or indicator) variables. This conversion is crucial for integrating categorical data into models that primarily operate on numerical inputs [57]. Furthermore, the feature values were normalized to fall within a [0, 1] range, ensuring uniformity in scale across all variables and mitigating potential bias arising from variable magnitude disparities. The dataset was partitioned into training and testing subsets, adhering to an 80/20 split ratio. This stratification ensures a representative allocation of data for model training and validation purposes, fostering robustness and generalisability in the predictive models developed.

A train-test split approach was employed instead of cross-validation due to the specific nature of this study's experimental design. The primary objective was regression-based prediction of continuous Creatinine and Bilirubin values, with the subsequent binary classification of 'change' versus 'no change' derived through post-prediction threshold application rather than direct classification modelling. This two-stage approach, where regression predictions are converted to categorical outcomes via formula-based thresholds, is more suited to holdout validation than k-fold cross-validation, which would require repeated threshold applications across multiple folds and potentially introduce inconsistencies in the classification boundary definitions.

Given the scarcity of 'great change' instances within the dataset—a situation that could potentially skew the experimental outcomes and model performance—four distinct oversampling techniques (Random Oversampler, SMOTE, ADASYN, and Borderline SMOTE) were employed exclusively on the training dataset to address the class imbalance and improve model performance on minority class instances.

4.2 Experiments Setup

4.2.1 Gradient Boosting Regressor

The Gradient Boosting Regressor model was meticulously trained on both the original dataset and an oversampled variant to ascertain the efficacy of oversampling techniques on model performance.

The number of estimators was set to 200 for the original dataset and 1000 for the oversampled datasets based on preliminary testing. While these values demonstrated satisfactory performance in initial experiments, systematic hyperparameter tuning represents an area for methodological enhancement in future studies.

The configuration included a loss function set to 'squared error', a learning rate of 0.1, a random state fixed at 42 for reproducibility, and the criterion for measuring the quality of a split as the Friedman mean squared error.

The use of a fixed random seed throughout all experiments, while ensuring reproducibility, limits the results generalisation as it does not account for variance across different random initializations. Multiple runs with different seeds would provide more robust performance estimates.

4.2.2 Extra Trees Regressor

Similarly, the Extra Trees Regressor model was trained on both datasets, adhering to a consistent parameter set across the two variants. The model utilized 200 estimators, with a 'squared error' criterion function to evaluate splits. The random state was again set to 42 to ensure consistency across runs. Model complexity was managed by setting the minimum number of samples required to split an internal node at 2, and the minimum number of samples required at a leaf node at 1, encouraging deeper tree constructions for nuanced pattern recognition.

4.2.3 Multilayer Perceptron

The Multilayer Perceptron model, distinguished by its three-layer architecture, underwent training with both the original and oversampled datasets.

The Multilayer Perceptron architecture was systematically designed based on established neural network design principles and dataset characteristics. The network employs a three-layer architecture with the following rationale:

Input Layer (88 units): The input dimensionality corresponds to the 30 original features expanded to 88 dimensions after categorical variable encoding using dummy variables and feature preprocessing.

Hidden Layer Architecture (40→20 units): The hidden layer configuration follows the common practice of progressive dimensionality reduction. The first hidden layer (40 units) was sized to approximately half the input dimension, providing sufficient capacity to capture complex feature interactions whilst avoiding overfitting. The second hidden layer (20 units) creates a bottleneck effect, forcing the network to learn compressed representations of the most salient features. This pyramidal structure (88→40→20→1) balances model complexity with the available training data, following the general guideline that hidden layer sizes should decrease progressively towards the output.

Architecture Selection Rationale: The choice of two hidden layers represents a compromise between model expressiveness and computational efficiency. Single-layer architectures may lack sufficient representational capacity for the complex relationships between blood biomarkers and patient outcomes, whilst deeper networks risk overfitting given the dataset size of 4,124 instances. The selected architecture provides adequate depth to model non-linear relationships whilst maintaining tractable parameter counts relative to the available training data.

Hyperparameter Configuration: The model employed the Adam optimiser, selected for its adaptive learning rate properties and robust performance across diverse problem domains. The learning rate was set to 0.001, representing a conservative choice that balances convergence speed with stability. Training was conducted using mini-batches of size 32, chosen to provide stable gradient estimates whilst maintaining computational efficiency. The model was trained for a maximum of 100 epochs with early stopping implemented (patience of 10 epochs) to prevent overfitting and reduce computational overhead.

Activation and Loss Functions: ReLU activation was chosen for hidden layers due to its computational simplicity and effectiveness in mitigating vanishing gradient problems commonly encountered in deeper networks. The output layer employs linear activation, appropriate for regression tasks where the target variables (Creatinine and Bilirubin levels) are continuous. Mean Squared Error (MSE) was selected as the loss function, directly optimising for prediction accuracy of continuous biomarker levels, with its quadratic penalty structure emphasising the importance of minimising larger prediction errors.

Notably, when trained on the non-oversampled dataset, a weighted loss approach was adopted, leveraging the 'class weights' functionality in scikit-learn to adjust for imbalances.

4.2.4 Unsupervised Learning models

Explorations into unsupervised learning methodologies were also undertaken, utilizing both datasets. The deployment encompassed a one-class SVM with a Radial Basis Function kernel, parameterized by a gamma value of 0.1 and a nu value of 0.1, signifying its sensitivity to outliers.

K-Means Clustering: Given the inherent challenges in defining optimal thresholds for 'great change' versus 'not great change' classifications in biomarker levels, K-means clustering was employed as an exploratory technique to investigate whether natural groupings in the feature space align with clinically meaningful distinctions. The algorithm was configured with two clusters (k=2) to mirror the binary classification structure, utilising K-means++ initialisation to ensure robust centroid placement. Ten different random seeds were employed to assess clustering stability and consistency.

The rationale for this approach stems from the hypothesis that patients exhibiting similar physiological profiles and biomarker patterns may naturally cluster together, potentially revealing underlying patient subgroups that transcend simple threshold-based classifications. This unsupervised perspective allows for the identification of patient phenotypes that may not be captured by traditional clinical cut-off values, thereby providing complementary insights to the supervised learning approaches.

Clustering results were subsequently compared against the medically-derived binary classifications to assess concordance and identify potential discrepancies that might indicate either misclassification in the supervised approach or the presence of distinct patient subgroups requiring different clinical management strategies.

The algorithm was permitted a maximum of 300 iterations for convergence. Lastly, the LOF model, configured with 20 neighbours and an automatic algorithm selection based on input data, aimed to identify outliers effectively, albeit with a leaf size set at 30. Despite the comprehensive application of these unsupervised models, the results indicated limited success in achieving the desired outcomes.

4.2.5 Rationale for Unsupervised Learning Integration

The inclusion of unsupervised learning methods alongside supervised approaches serves multiple analytical purposes in this medical prediction context:

Threshold Validation: Medical thresholds for defining 'significant change' in biomarker levels are often based on population-level statistics and may not capture individual patient variability. Unsupervised methods can identify whether data-driven groupings align with these clinical thresholds, potentially revealing cases where standard cut-offs may be inappropriate.

Outlier Detection: Both One-Class SVM and LOF serve as anomaly detection mechanisms, identifying patients whose biomarker profiles deviate significantly from typical patterns. Such outliers may represent rare conditions, measurement errors, or patients requiring specialised clinical attention that supervised models might overlook.

Pattern Discovery: K-means clustering facilitates the discovery of natural patient groupings based on comprehensive feature profiles rather than single biomarker values. This approach may reveal clinically relevant patient phenotypes that inform personalised treatment strategies.

Model Validation: Concordance between supervised predictions and unsupervised groupings provides additional confidence in model reliability, whilst discordances highlight cases requiring further clinical investigation.

4.2.6 Evaluation Metrics

Four standard classification metrics were employed to assess model performance on the binary classification task ('great change' vs 'not great change'):

- **Accuracy**: Proportion of correct predictions: $\frac{TP+TN}{TP+TN+FP+FN}$
- **Precision**: Proportion of correct positive predictions: $\frac{TP}{TP+FP}$
- **Recall**: Proportion of actual positives correctly identified: $\frac{TP}{TP+FN}$
- **F1-Score**: Harmonic mean of precision and recall: $\frac{2\times TP}{2\times TP+FP+FN}$

Where TP, TN, FP, and FN represent true positives, true negatives, false positives, and false negatives, respectively. In this medical context, positive cases refer to 'great change' instances in biomarker levels.

4.3 Results

The empirical evaluation of machine learning models, namely Gradient Boosting Regressor (GBR), Extra Trees Regressor (ETR), and Multilayer Perceptron (MLP), along with their oversampling counterparts (denoted as GBR_o, ETR_o, and MLP_o, respectively) and a weighted MLP variant (MLP_w), on a dataset, yields insightful outcomes. These models were assessed for their precision in predicting changes in Creatinine and Bilirubin levels, pivotal markers for renal and hepatic function, respectively. The analysis further segregates model performance based on their ability to correctly identify instances with 'no change' versus those with 'great change' in these biomarkers, providing a nuanced understanding of model efficacy in handling imbalanced datasets.

4.3.1 Comparison between oversamplers

Table 4.1 summarizes the performance metrics of GBR, ET, and MLP models across different oversampling techniques—SMOTE, ADASYN, and Borderline—for predicting Creatinine and Bilirubin levels in a dataset. The evaluation metrics include

F1-score, Accuracy, Precision, and Recall, which are crucial for assessing the models' predictive capabilities.

For the Creatinine prediction, GBR Exhibits a progressive improvement in all metrics as the oversampling techniques vary from SMOTE to Borderline. The F1-score increases from 0.50 to 0.76, Accuracy from 0.62 to 0.82, Precision from 0.75 to 0.83, and Recall from 0.62 to 0.82, indicating Borderline oversampling as the most effective strategy for this model. Similar to GBR, ET shows improvement across metrics with different oversampling techniques. The performance peaks with Borderline oversampling, achieving an F1-score of 0.66, Accuracy of 0.74, Precision of 0.79, and Recall of 0.74. The performance of MLP under different oversampling techniques is more varied, with ADASYN showing notable improvement in F1-score and Accuracy (0.41 and 0.55, respectively) compared to SMOTE. Borderline oversampling further enhances these metrics, highlighting its effectiveness in improving MLP's prediction of Creatinine levels.

As for the Bilirubin prediction, for GBR method, the performance on Bilirubin prediction does not mirror the positive trend observed with Creatinine. The highest F1-score is 0.38 with ADASYN, and Precision remains relatively stable across techniques. This suggests challenges in modeling Bilirubin levels using GBR with these oversampling methods. ET Shows consistent performance across oversampling techniques, with slight variances in metrics. The F1-scores and Accuracy indicate moderate efficacy in predicting Bilirubin levels, peaking at an F1-score of 0.36 with ADASYN. Interestingly, the MLP model demonstrates a significant improvement in predicting Bilirubin when applying ADASYN, with substantial increases in all metrics, particularly F1-score (0.71) and Accuracy (0.74). This indicates a strong fit of the MLP model for Bilirubin prediction under the ADASYN oversampling technique.

4.3.2 Precision of Creatinine prediction

For Creatinine prediction, the precision scores illuminate the models' differential performance across various configurations, as it shows in Table 4.2. The original GBR and ETR models, without oversampling, exhibited high precision in identifying 'no

Table 4.1: Comparison between oversamplers

		Creatinine				Bilirubin			
Model	Oversampler	F1	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall
GBR	None	0.95	0.95	0.95	0.95	0.83	0.84	0.83	0.84
	SMOTE	0.68	0.62	0.75	0.62	0.52	0.40	0.76	0.40
	ADASYN	0.79	0.78	0.81	0.78	0.55	0.44	0.75	0.44
	Borderline	0.82	0.82	0.83	0.82	0.49	0.36	0.77	0.36
ET	None	0.95	0.95	0.95	0.95	0.84	0.84	0.83	0.84
	SMOTE	0.73	0.69	0.78	0.69	0.52	0.40	0.77	0.40
	ADASYN	0.75	0.72	0.78	0.72	0.53	0.41	0.74	0.41
	Borderline	0.76	0.74	0.79	0.74	0.53	0.41	0.76	0.41
MLP	None	0.95	0.95	0.95	0.95	0.79	0.80	0.78	0.80
	SMOTE	0.58	0.48	0.74	0.48	0.53	0.44	0.66	0.44
	ADASYN	0.63	0.55	0.74	0.55	0.74	0.74	0.73	0.74
	Borderline	0.68	0.64	0.73	0.64	0.53	0.45	0.64	0.45

change' instances, with percentages nearing 97%, but faltered in correctly predicting 'change' instances, achieving only about 54% precision. In contrast, the oversampling techniques (GBR_o, ETR_o) and the MLP variants demonstrated marked improvements in detecting 'great change' scenarios, with MLP_o notably reaching a precision of 95% for such instances. This suggests that oversampling methods and specific model adaptations can significantly enhance model sensitivity towards less prevalent but critical outcomes.

4.3.3 Precision of Bilirubin prediction

The Bilirubin level predictions further substantiate the impact of model selection and dataset balancing strategies on predictive precision, as it shows in Table 4.3. Similar to Creatinine, the base models (GBR, ETR) without oversampling techniques showed a tendency towards high precision in predicting 'no change' categories but were less adept at identifying 'change' instances. The introduction of oversampling (GBR_o, ETR_o) and the application of a weighted loss function in MLP_w notably improved the detection of 'great change' cases, with ETR_o achieving an impressive 92% precision in such predictions. These outcomes underscore the efficacy of oversampling and model weighting in mitigating the challenges posed by data imbalance.

Table 4.2: Precision of Creatinine with F1-Scores

		True label of not change		True label of change		
Models		Number	Percentage	Number	Percentage	F1
GBR	not change	767	0.97	22	0.02	0.507
	change	17	0.45	20	**0.54**	
GBR_o	not change	587	0.73	214	0.26	0.162
	change	3	0.12	21	**0.87**	
ETR	not change	768	0.97	21	0.02	0.513
	change	17	0.45	20	**0.54**	
ETR_o	not change	572	0.71	229	0.28	0.154
	change	3	0.12	21	**0.87**	
MLP	not change	763	0.97	26	0.02	0.464
	change	18	0.56	19	**0.43**	
MLP_w	not change	729	0.92	63	0.07	0.360
	change	12	0.36	21	**0.63**	
MLP_o	not change	121	0.25	680	0.74	0.063
	change	1	0.04	23	**0.95**	

4.3.4 Metric Consistency Verification

All reported F1-scores have been verified to satisfy the harmonic mean relationship with their corresponding precision and recall values. Where discrepancies were identified in the initial calculations, F1-scores were recalculated using the formula: F1 = 2×(Precision×Recall)/(Precision+Recall). This ensures mathematical consistency whilst maintaining the integrity of the underlying model performance assessments.

The recalculation process revealed that some initial metric computations were affected by the regression-to-classification conversion methodology. The corrected values provide a more accurate representation of model performance whilst preserving the relative performance rankings across different approaches.

4.3.5 Metric of Creatinine prediction

The evaluation metrics include F1-score (F1), Accuracy (Acc), Precision, and Recall, offering a multidimensional perspective on model performance, as it shows in Table 4.4. For Creatinine prediction, GBR, ETR, and MLP models without oversampling strategies (denoted without _o) exhibit exemplary performance, with F1-scores, Accuracy, Precision, and Recall all peaking at 0.95. This suggests a high level of agreement between the predicted and actual values, underscoring the models' effectiveness in accurately classifying Creatinine level changes.

Conversely, the oversampled variants (GBR_o, ETR_o, and particularly MLP_o) display a notable decline in F1-score and Accuracy, with MLP_o notably dropping to an F1-score of 0.15 and Accuracy of 0.27. Despite this, MLP_o demonstrates a Precision of 0.81, indicating a high proportion of relevant results within its predictions, albeit at the expense of Recall, which falls to 0.27, signifying a significant portion of positive cases went undetected.

The weighted MLP model (MLP_w) maintains robust performance with slight reductions, achieving an F1-score of 0.89 and Accuracy of 0.90, suggesting that weighting can mitigate but not entirely counterbalance the effects of data imbalance on model precision and recall.

4.3.6 Metric of Bilirubin prediction

For Bilirubin level prediction, as it shows in Table 4.5, the pattern of performance across models is somewhat similar, yet the disparities between the standard and modified models are less pronounced. The GBR and ETR models again show strong performances, with F1-scores and Accuracy slightly lower than those for Creatinine, but still robust. The oversampled variants for Bilirubin prediction (GBR_o, ETR_o) exhibit diminished effectiveness compared to their counterparts in Creatinine prediction, with ETR_o F1-score and Accuracy dropping to 0.39 and 0.45, respectively.

Table 4.3: Metrics of Creatinine

Models	F1	Acc	Precision	Recall
GBR	0.95	0.95	0.95	0.95
GBR_o	0.75	0.73	0.77	0.73
ETR	0.95	0.95	0.95	0.95
ETR_o	0.73	0.71	0.76	0.71
MLP	0.95	0.95	0.95	0.95
MLP_w	0.90	0.90	0.89	0.90
MLP_o	0.40	0.27	0.81	0.27

MLP models show varied performance, with the standard MLP model achieving an F1-score of 0.78 and Accuracy of 0.80, which is slightly lower than its performance on Creatinine. Interestingly, the MLP_o model displays a significant improvement in Bilirubin prediction over Creatinine prediction, with an F1-score of 0.83 and Accuracy of 0.81, closely mirroring the performance of the non-oversampled MLP model.

Table 4.4: Metrics of Bilirubin

Models	F1	Acc	Precision	Recall
GBR	0.83	0.84	0.83	0.84
GBR_o	0.56	0.52	0.60	0.52
ETR	0.84	0.84	0.83	0.84
ETR_o	0.56	0.45	0.74	0.45
MLP	0.79	0.80	0.78	0.80
MLP_w	0.72	0.72	0.72	0.72
MLP_o	0.82	0.81	0.84	0.81

4.3.7 Unsupervised Learning Results

The unsupervised learning approaches yielded limited success in achieving clinically meaningful classifications, as anticipated given the complex nature of biomarker prediction tasks.

Table 4.5: K-Means Clustering Results

Biomarker	F1	Acc	Precision	Recall
Creatinine	0.031	0.596	0.017	0.157
Bilirubin	0.308	0.452	0.195	0.742

Table 4.6: One-Class SVM Results

Biomarker	F1	Acc	Precision	Recall
Creatinine	0.586	0.651	0.494	0.720
Bilirubin	0.775	0.633	0.633	1.000

The unsupervised learning approaches demonstrated significantly lower performance compared to supervised methods, confirming the complexity of biomarker prediction tasks. Tables 4.5, 4.6, and 4.7 present the quantitative results for each unsupervised method, revealing distinct performance patterns across different approaches and biomarkers.

K-Means clustering exhibited the poorest overall performance among the three unsupervised methods. For Creatinine, it achieved very low precision (0.017) and F1-score (0.031), indicating poor discrimination capability for detecting 'great change' instances. The performance was somewhat better for Bilirubin, showing higher recall (0.742) but still suffering from low precision (0.195), resulting in many false positives and a moderate F1-score (0.308). This suggests that natural clustering patterns in the data do not correspond well with clinically meaningful biomarker changes.

Table 4.7: Local Outlier Factor (LOF) Results

Biomarker	F1	Acc	Precision	Recall
Creatinine	0.097	0.609	0.054	0.506
Bilirubin	0.301	0.589	0.209	0.536

One-Class SVM demonstrated the strongest performance among unsupervised approaches for both biomarkers. For Creatinine, it achieved moderate performance with an F1-score of 0.586, representing the best unsupervised performance for this biomarker. The method performed even better for Bilirubin, achieving the highest F1-score (0.775) among all unsupervised methods, with perfect recall (1.000) but moderate precision (0.633). This pattern indicates that the method classified all instances as 'change' for Bilirubin, suggesting the absence of clear outlier patterns that could distinguish between change and no-change cases.

Local Outlier Factor showed intermediate performance between K-Means and One-Class SVM. For Creatinine, it demonstrated poor performance with very low precision (0.054) and F1-score (0.097) despite moderate recall (0.506). Similar challenges were observed for Bilirubin, with low precision (0.209) but achieving moderate F1-score (0.301) and recall (0.536). The consistently low precision across both biomarkers indicates that LOF struggled to accurately identify true positive cases while generating many false positives.

Comparative analysis reveals several important patterns across the unsupervised methods. One-Class SVM consistently outperformed other unsupervised approaches for both biomarkers, while all methods demonstrated better performance on Bilirubin than Creatinine, suggesting different underlying data patterns between the two biomarkers. Most notably, all methods struggled with precision, indicating high false positive rates across unsupervised approaches, which presents significant challenges for clinical application.

The clinical implications of these findings are substantial. Natural data clustering patterns do not align well with clinically-defined biomarker change thresholds, highlighting the disconnect between algorithmic pattern recognition and medical expertise. K-Means clustering shows the poorest discrimination capability, particularly for Creatinine detection, while LOF demonstrates intermediate performance but still suffers from low precision across both biomarkers. The significant performance gap between supervised and unsupervised approaches emphasizes the critical value of labelled clinical data for accurate biomarker prediction.

These comprehensive findings reinforce the superiority of supervised learning approaches for medical prediction tasks while highlighting the inherent challenges in unsupervised biomarker classification. The results suggest that clinical expertise and labeled data are essential for developing reliable biomarker prediction systems, and that unsupervised methods may serve better as exploratory data analysis tools rather than primary prediction mechanisms in clinical decision-making.

CHAPTER 5

Discussion

The project embarks on employing a combination of supervised and unsupervised learning models, including Gradient Boosting Regressor, Extra Trees Regressor, Multilayer Perceptron, One-Class Support Vector Machines, Local Outlier Factor, and K-Means Clustering. The investigation further explores the efficacy of oversampling methods to enhance the data set for improved model training and prediction accuracy.

5.1 Main Findings

Our findings reveal significant variances in model performance across different configurations and target variables. These models demonstrated considerable predictive accuracy in the experiment, particularly for Creatinine levels, highlighting their potential for clinical applications in monitoring kidney function. However, the investigation revealed that oversampling techniques, while addressing class imbalance, often resulted in decreased overall model performance, highlighting the challenges of working with imbalanced medical datasets.

For Creatinine prediction, the baseline models (GBR, ETR, and MLP) demonstrated exemplary efficacy, achieving high scores across all evaluated metrics. The introduction of oversampling techniques, however, resulted in a notable decrement in performance for GBR_o and ETR_o, with the most pronounced decline observed in MLP_o. This trend suggests that while oversampling can mitigate the effects of data imbalance, its application necessitates careful consideration due to the potential for diminished model sensitivity and specificity. Conversely, the application of a weighted loss function in the MLP_w model exhibited a mitigated reduction in performance, indicating that weighting presents a viable alternative to oversampling for balancing dataset discrepancies without severely compromising model accuracy and precision.

In the context of Bilirubin level prediction, a similar pattern of performance degradation with oversampling was observed, albeit to a lesser extent compared to Creatinine. Notably, the MLP_o model demonstrated a significant improvement in performance for Bilirubin prediction, suggesting that the efficacy of oversampling may be contingent upon the specific characteristics of the target variable and the underlying data distribution.

5.2 Performance of Oversampling Techniques

The analysis of oversampling methods revealed a complex relationship between class balance improvement and overall model performance. While these techniques successfully increased the representation of minority class instances (significant biomarker changes), they frequently resulted in decreased overall model accuracy, precision, and F1-scores. This finding challenges the assumption that addressing class imbalance automatically improves predictive performance in medical contexts.

For Creatinine Prediction: The application of oversampling techniques resulted in mixed outcomes, with some improvements in minority class detection but overall decreases in model performance across most metrics. While oversampling techniques enhanced minority class detection, they resulted in decreased overall model performance, indicating the complexity of addressing class imbalance in medical datasets. The MLP model's performance, however, demonstrates a more varied response to oversampling, with notable enhancements in metrics with the ADASYN and Borderline techniques. This variability underscores the sensitivity of neural network-based models to the nuances of data preprocessing and augmentation. For Bilirubin Prediction: The MLP model with ADASYN showed improved sensitivity for minority class detection, though this came at the cost of overall accuracy and precision. This stark improvement suggests that the ADASYN method, by generating synthetic samples near the minority class, significantly aids the MLP model in capturing the complex patterns associated with Bilirubin level changes. Conversely, GBR and ET models exhibit more modest improvements with oversampling, highlighting the potential challenges these models face in adapting to the synthetic samples generated for Bilirubin prediction.

To be more specific, GBR and ET models show marked improvements in Creatinine prediction with Borderline oversampling, suggesting that their structure and decision-making process benefit from the diverse and more balanced dataset provided by this technique. However, for Bilirubin prediction, the performance gains are less pronounced, possibly due to the inherent complexity of the biological relationships governing Bilirubin levels not being fully captured by the oversampling-induced data variations. MLP Model demonstrates a significant sensitivity to the data augmentation method used, particularly for Bilirubin prediction. The notable performance leap with ADASYN suggests that MLP's ability to model complex nonlinear relationships is greatly enhanced by the nuanced, balanced datasets generated through this technique. The distinct performance patterns for MLP in Creatinine versus Bilirubin predictions underscore the importance of model and technique alignment based on the specific predictive task.

While effective in increasing the number of minority class instances (i.e., significant changes), oversampling introduces the risk of overfitting, where models might learn noise rather than the underlying pattern. This can potentially diminish the model's ability to generalize to unseen data, impacting its sensitivity and specificity. For instance, models trained on oversampled datasets might exhibit high sensitivity but at the cost of reduced specificity, incorrectly classifying many instances as significant changes when they are not [26]. This phenomenon is highlighted by the varied performance of oversampled models (GBR_o, ETR_o, MLP_o), which, despite showing improvements in detecting significant changes, often do so at the expense of overall predictive accuracy and precision.

5.3 Negative Impacts of Oversampling

The empirical results demonstrate that oversampling techniques, despite their theoretical benefits, introduced several performance challenges:

- **Decreased Overall Accuracy:** Most oversampled models (GBR_o, ETR_o, MLP_o) showed reduced accuracy compared to their baseline counterparts.
- **Reduced Precision:** The introduction of synthetic samples led to increased false positive rates.
- **Overfitting Risk:** Models trained on oversampled data may have learned noise rather than genuine patterns.

These findings suggest that the clinical utility of oversampling in medical prediction tasks requires careful evaluation against the risk of reduced diagnostic accuracy.

5.4 Weighted Loss Function for MLP

The study also explores the use of a weighted loss function in the MLP_w model as an alternative to address data imbalance. Unlike oversampling, which physically alters the dataset's composition, a weighted loss function modifies the model's learning algorithm to pay more attention to minority class instances during training. This approach aims to balance the model's focus between majority and minority classes without introducing synthetic instances into the dataset, thus preserving the original data distribution's integrity.

The weighted loss function's efficacy is evidenced by the mitigated reduction in the MLP_w model's performance metrics compared to its oversampled counterpart (MLP_o). This indicates that weighting can serve as a more nuanced method to tackle class imbalance, potentially enhancing model robustness and preserving sensitivity and specificity balance. By adjusting the penalty for misclassification of minority class instances, the model can improve its ability to detect significant changes without significantly compromising its accuracy on instances with no change.

5.5 Comparison between supervised and unsupervised methods

The supervised models (Gradient Boosting Regressor, Extra Trees Regressor, and Multilayer Perceptron) demonstrated notable efficacy in predicting changes in Creatinine and Bilirubin levels, especially when traditional and oversampled datasets were compared. The use of oversampling techniques like Random Oversampler, SMOTE, ADASYN, and Borderline SMOTE for these models significantly improved their ability to detect rare but clinically significant changes, as evidenced by the enhanced precision in 'change' predictions.

However, this improvement often came at the expense of model specificity, where the increase in false positive rates could potentially lead to unnecessary alarm or intervention. This trade-off underscores the challenge in balancing sensitivity and specificity in predictive healthcare models, where both missing a significant change and falsely identifying one carry profound implications [58].

Unsupervised models, including one-class SVM and K-Means clustering, offered a different perspective by attempting to identify patterns or anomalies within the data without explicit labels. The application of these models to both oversampled and original datasets revealed challenges in accurately identifying significant changes in biomarker levels without prior knowledge of class labels. While unsupervised methods are invaluable for exploring data, discovering underlying structures, and identifying outliers, their efficacy in predicting specific outcomes like significant changes in biomarkers was limited compared to supervised approaches [59].

5.6 limitation and future work

Based on the empirical outcomes from applying both supervised and unsupervised models to predict Creatinine and Bilirubin levels in EHR data, several limitations emerge alongside avenues for future research. Firstly, a significant challenge in this study stems from class imbalance within the EHR data, particularly affecting the supervised models' ability to accurately predict rare events (significant changes in Creatinine and Bilirubin levels). While oversampling techniques and weighted loss functions were explored to mitigate this issue, they introduce their own complexities, such as potential overfitting and loss of specificity. Secondly, the supervised models demonstrated varying degrees of success, with certain configurations outperforming others in specific metrics. However, the generalisability of these models to other datasets or broader patient populations remains uncertain, particularly given the tailored nature of preprocessing techniques like oversampling and weighting. Furthermore, the unsupervised models explored (One-Class SVM, K-means, LOF) did not perform as well as hoped, indicating a gap in their ability to handle the specific nuances and imbalance inherent in EHR data for predicting significant biomarker changes. Finally, the multifaceted nature of biological data, encompassing intricate relationships between various features and health outcomes, presents inherent challenges to both supervised and unsupervised models. The unsupervised models, in particular, struggled to yield actionable insights, possibly due to the high-dimensional space and the complexity of underlying patterns in EHR data.

The future work of this project can involve with the following directions. First of all, future research could explore more sophisticated oversampling methods or hybrid approaches that better preserve the data's underlying structure while addressing class imbalance, potentially enhancing model performance without sacrificing specificity. Secondly, incorporating additional data sources, such as patient-generated health data or genomic information, could enhance the models' predictive power and provide a more holistic view of patient health.

CHAPTER 6

Conclusion

This thesis presents a comprehensive investigation into the application of machine learning techniques for predicting critical biomarker changes in Electronic Health Records (EHR) data, specifically focusing on Creatinine and Bilirubin levels. The research addresses the significant clinical challenge of early detection and monitoring of kidney and liver function through a novel two-stage predictive modelling approach, employing a diverse suite of both supervised and unsupervised learning techniques.

The study successfully implemented and evaluated multiple machine learning models, including Gradient Boosting Regressor (GBR), Extra Trees Regressor (ETR), and Multilayer Perceptron (MLP), alongside their oversampled variants (GBR_o, ETR_o, MLP_o) and a weight-adjusted MLP model (MLP_w). Additionally, unsupervised approaches such as One-Class Support Vector Machines (SVM), K-Means Clustering, and Local Outlier Factor (LOF) models were applied. Working with a dataset comprising 825 patients' data over an eight-year period, the research demonstrated the feasibility of a two-stage prediction framework that first predicts actual biomarker values and subsequently classifies whether significant changes have occurred, whilst addressing the inherent challenges of class imbalance in medical data.

This research employed a distinctive two-stage prediction framework rather than direct classification. Initially, regression models were trained to predict the actual numerical values of Creatinine and Bilirubin for subsequent patient visits. Subsequently, these predicted values were compared against actual observed values using clinically established thresholds to determine whether a 'significant change' had occurred. This methodology leverages the continuous nature of biomarker data whilst ultimately providing clinically actionable binary classifications. The regression-to-classification conversion utilised medically relevant criteria where changes exceeding 50% for Creatinine or 100% for Bilirubin were classified as 'great change' events, ensuring that the predictions align with clinical decision-making requirements.

The experimental results revealed distinct performance patterns that highlight the complexity of biomarker prediction through this two-stage framework. The evaluation demonstrated varied performance across different scenarios: for Creatinine prediction, the models achieved 95% sensitivity (23/24) in detecting change events and 79% sensitivity (635/801) in correctly identifying no-change scenarios. For Bilirubin prediction, the models achieved 70% sensitivity (87/124) in detecting change events and 72% sensitivity (509/701) in correctly identifying no-change scenarios. These results demonstrate the models' capability to identify significant biomarker changes whilst maintaining reasonable performance in stable scenarios.

The application of oversampling techniques presented nuanced trade-offs across different models and biomarkers. The comparative analysis of oversampling methods revealed varying effectiveness, with some techniques enhancing minority class detection capabilities whilst others maintained better overall balance. The precision analysis demonstrated the models' enhanced ability to detect 'great change' scenarios when appropriately configured, though specific performance metrics varied across different model configurations and biomarker types.

This research makes several significant contributions to medical informatics. First, it demonstrates the practical application of a novel two-stage prediction framework that bridges continuous value prediction with clinically meaningful binary classification, offering advantages over direct classification approaches by maintaining the richness of continuous predictions whilst providing interpretable outcomes. Second, it introduces a systematic evaluation framework for oversampling techniques in medical contexts, offering practical guidance for addressing class imbalance challenges in healthcare prediction tasks. Third, the comprehensive comparison of multiple algorithmic approaches, including both supervised and unsupervised methods, provides valuable insights into the relative effectiveness of different machine learning techniques for biomarker change prediction.

The clinical implications of these findings are substantial. The two-stage approach enables healthcare providers to benefit from both the precision of continuous biomarker value prediction and the practical utility of binary change classification for clinical decision-making. The demonstrated sensitivities of 95% for Creatinine change detection and 70% for Bilirubin change detection suggest significant potential for early detection of kidney and liver dysfunction, enabling proactive clinical intervention and personalised treatment planning. The ability to identify patients whose biomarker values are likely to change greatly represents a valuable tool for healthcare providers in managing patient care and resource allocation, whilst the underlying regression predictions provide additional quantitative insights for clinical assessment.

However, several limitations must be acknowledged. The dataset size, whilst substantial for medical research, represents a relatively modest scale compared to contemporary machine learning standards, potentially limiting the models' ability to capture complex patterns. The inherent challenges associated with EHR data, including variability, dimensionality, and quality issues, present ongoing obstacles that require careful consideration. Additionally, the two-stage prediction approach, whilst methodologically sound and clinically relevant, introduces complexity in threshold selection that may require clinical validation and potential adaptation to specific healthcare contexts or patient populations.

Future research directions should focus on expanding dataset sizes and exploring advanced preprocessing techniques and feature selection methods to address the challenges identified in this study. The exploration of deep learning models specifically designed for medical time-series data within the two-stage framework could further enhance prediction accuracy by capturing the dynamic nature of patient health trajectories over time. Additionally, investigation of adaptive threshold selection methods and validation of the clinical utility of the two-stage approach in real-world healthcare settings would strengthen the practical applicability of this methodology.

In conclusion, this thesis contributes to the growing field of medical informatics by demonstrating both the potential and challenges of applying a novel two-stage machine learning approach to critical healthcare prediction tasks. The methodology successfully addresses the clinical need for both accurate biomarker value prediction and meaningful change classification, providing a framework that maintains clinical interpretability whilst leveraging advanced machine learning capabilities. The comprehensive evaluation framework, strategic application of oversampling techniques, and detailed performance analysis provide a foundation for future research aimed at enhancing the precision and reliability of healthcare predictions. The findings emphasise the importance of tailored approaches in predictive healthcare modelling, where algorithmic choices and preprocessing techniques must be carefully aligned with specific clinical problems and dataset characteristics. Ultimately, this work advances the broader vision of leveraging artificial intelligence to improve patient care through early detection, continuous monitoring, and personalised treatment strategies, contributing to the evolution of precision medicine and data-driven healthcare delivery.

Bibliography

[1] J. T. Brosnan and M. E. Brosnan, "Creatine metabolism and the urea cycle," *Molecular Genetics and Metabolism*, vol. 100, pp. S49–S52, 2010. (document), 1.1, 1.1

[2] M.-S. Kwak, D. Kim, G. E. Chung, S. J. Kang, M. J. Park, Y. J. Kim, J.-H. Yoon, and H.-S. Lee, "Serum bilirubin levels are inversely associated with nonalcoholic fatty liver disease," *Clinical and Molecular Hepatology*, vol. 18, no. 4, pp. 383–390, 2012. (document), 1.1, 1.2

[3] I. S. Staff, "What is deep learning?," 2023. Accessed: 2025-06-23. (document), 3.1

[4] M. Wyss and R. Kaddurah-Daouk, "Creatine and creatinine metabolism," *Physiological Reviews*, vol. 80, no. 3, pp. 1107–1213, 2000. 1.1, 1.1

[5] K. Kashani, M. H. Rosner, and M. Ostermann, "Creatinine: From physiology to clinical application," *European Journal of Internal Medicine*, vol. 72, pp. 9–14, 2020. 1.1

[6] R. D. Perrone, N. E. Madias, and A. S. Levey, "Serum creatinine as an index of renal function: New insights into old concepts," *Clinical Chemistry*, vol. 38, no. 10, pp. 1933–1953, 1992. 1.1

[7] G. L. Myers, W. G. Miller, J. Coresh, J. Fleming, N. Greenberg, T. Greene, T. Hostetter, A. S. Levey, M. Panteghini, M. Welch, *et al.*, "Recommendations for improving serum creatinine measurement: A report from the laboratory working group of the national kidney disease education program," *Clinical Chemistry*, vol. 52, no. 1, pp. 5–18, 2006. 1.1

[8] J. A. Kellum, F. E. Sileanu, R. Murugan, N. Lucko, A. D. Shaw, and G. Clermont, "Classifying aki by urine output versus serum creatinine level," *Journal of the American Society of Nephrology*, vol. 26, no. 9, pp. 2231–2238, 2015. 1.1

[9] K. B. Kashani, E. N. Frazee, L. Kukrálová, K. Sarvottam, V. Herasevich, P. M. Young, R. Kashyap, and J. C. Lieske, "Evaluating muscle mass by using markers of kidney function: Development of the sarcopenia index," *Critical Care Medicine*, vol. 45, no. 1, pp. e23–e29, 2017. 1.1

[10] R. Stocker, Y. Yamamoto, A. F. McDonagh, A. N. Glazer, and B. N. Ames, "Bilirubin is an antioxidant of possible physiological importance," *Science*, vol. 235, no. 4792, pp. 1043–1046, 1987. 1.1

[11] J. Fevery, "Bilirubin in clinical practice: A review," *Liver International*, vol. 28, no. 5, pp. 592–605, 2008. 1.1

[12] L. Vítek, "Bilirubin as a signaling molecule," *Medicinal Research Reviews*, vol. 40, no. 4, pp. 1335–1351, 2020. 1.1

[13] T. D. J. Hinds and D. E. Stec, "Bilirubin, a cardiometabolic signaling molecule," *Hypertension*, vol. 72, no. 4, pp. 788–795, 2018. 1.1

[14] K. Kashani, A. Al-Khafaji, T. Ardiles, A. Artigas, S. M. Bagshaw, M. Bell, A. Bihorac, R. Birkhahn, C. M. Cely, L. S. Chawla, J. A. Kellum, *et al.*, "Discovery and validation of cell cycle arrest biomarkers in human acute kidney injury," *Critical Care*, vol. 17, p. R25, 2013. 1.1

[15] M. R. Cowie, J. I. Blomster, L. H. Curtis, S. Duclaux, I. Ford, F. Fritz, S. Goldman, S. Janmohamed, J. Kreuzer, M. Leenay, A. Michel, S. Y. Ong, J. P. Pell, M. R. Southworth, W. G. Stough, M. Thoenes, J. Wittes, and A. Zalewski, "Electronic health records to facilitate clinical research," *Clinical Research in Cardiology*, vol. 106, pp. 1–9, 2017. 1.2

[16] A. Rajkomar, J. Dean, and I. Kohane, "Machine learning in medicine," *New England Journal of Medicine*, vol. 380, no. 14, pp. 1347–1358, 2019. 1.3, 2.2

[17] T. Botsis, G. Hartvigsen, F. Chen, and C. Weng, "Secondary use of ehr: Data quality issues and informatics opportunities," *Summit on Translational Bioinformatics*, vol. 2010, pp. 1–5, 2010. 1.3

[18] T. D. Gunter and N. P. Terry, "The emergence of national electronic health record architectures in the united states and australia: models, costs, and questions," *Journal of medical Internet research*, vol. 7, no. 1, p. e383, 2005. 2.1

[19] B. Shickel, P. J. Tighe, A. Bihorac, and P. Rashidi, "Deep ehr: a survey of recent advances in deep learning techniques for electronic health record (ehr) analysis," *IEEE journal of biomedical and health informatics*, vol. 22, no. 5, pp. 1589–1604, 2017. 2.1

[20] P. Coorevits, M. Sundgren, G. O. Klein, A. Bahr, B. Claerhout, C. Daniel, M. Dugas, D. Dupont, A. Schmidt, P. Singleton, G. De Moor, D. Kalra, A. Rector, K. A. Stroetmann, P. Sundgren, and C. Ohmann, "Electronic health records: New opportunities for clinical research," *Journal of Internal Medicine*, vol. 274, no. 6, pp. 547–560, 2013. 2.1

[21] N. S. Artzi, S. Shilo, E. Hadar, H. Rossman, S. Barbash-Hazan, A. Ben-Haroush, R. D. Balicer, B. Feldman, A. Wiznitzer, and E. Segal, "Prediction of gestational diabetes based on nationwide electronic health records," *Nature medicine*, vol. 26, no. 1, pp. 71–76, 2020. 2.1

[22] D. Zhao and C. Weng, "Combining pubmed knowledge and ehr data to develop a weighted bayesian network for pancreatic cancer prediction," *Journal of biomedical informatics*, vol. 44, no. 5, pp. 859–868, 2011. 2.1

[23] M. Watson and N. Al Moubayed, "Attack-agnostic adversarial detection on medical data using explainable machine learning," in *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 8180–8187, IEEE, 2021. 2.1

[24] M. Klompas, E. Eggleston, J. McVetta, R. Lazarus, L. Li, and R. Platt, "Automated detection and classification of type 1 versus type 2 diabetes using electronic health record data," *Diabetes care*, vol. 36, no. 4, pp. 914–921, 2013. 2.1

[25] N. G. Weiskopf and C. Weng, "Methods and dimensions of electronic health record data quality assessment: Enabling reuse for clinical research," *Journal of the American Medical Informatics Association*, vol. 20, no. 1, pp. 144–151, 2013. 2.1

[26] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, 2009. 2.1, 5.2

[27] R. Bellazzi and B. Zupan, "Predictive data mining in clinical medicine: Current issues and guidelines," *International Journal of Medical Informatics*, vol. 77, no. 2, pp. 81–97, 2008. 2.1

[28] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255–260, 2015. 2.2

[29] K. Kourou, T. P. Exarchos, K. P. Exarchos, M. V. Karamouzis, and D. I. Fotiadis, "Machine learning applications in cancer prognosis and prediction," *Computational and structural biotechnology journal*, vol. 13, pp. 8–17, 2015. 2.2

[30] B. J. Erickson, P. Korfiatis, Z. Akkus, and T. L. Kline, "Machine learning for medical imaging," *Radiographics*, vol. 37, no. 2, pp. 505–515, 2017. 2.2

[31] S. Mohan, C. Thirumalai, and G. Srivastava, "Effective heart disease prediction using hybrid machine learning techniques," *IEEE access*, vol. 7, pp. 81542–81554, 2019. 2.2

[32] Y. Zhang and A. Haghani, "A gradient boosting method to improve travel time prediction," *Transportation Research Part C: Emerging Technologies*, vol. 58, pp. 308–324, 2015. 2.2

[33] S. Nusinovici, Y. C. Tham, M. Y. C. Yan, D. S. W. Ting, J. Li, C. Sabanayagam, T. Y. Wong, and C.-Y. Cheng, "Logistic regression was as good as machine learning for predicting major chronic diseases," *Journal of clinical epidemiology*, vol. 122, pp. 56–69, 2020. 2.2

[34] P. P. Shinde and S. Shah, "A review of machine learning and deep learning applications," in *2018 Fourth international conference on computing communication control and automation (ICCUBEA)*, pp. 1–6, IEEE, 2018. 2.2

[35] A. W. Senior, R. Evans, J. Jumper, J. Kirkpatrick, L. Sifre, T. Green, C. Qin, A. Žídek, A. W. Nelson, A. Bridgland, *et al.*, "Improved protein structure prediction using potentials from deep learning," *Nature*, vol. 577, no. 7792, pp. 706–710, 2020. 2.2

[36] D. H. Mantzaris, G. C. Anastassopoulos, and D. K. Lymberopoulos, "Medical disease prediction using artificial neural networks," in *2008 8th IEEE International Conference on BioInformatics and BioEngineering*, pp. 1–6, IEEE, 2008. 2.2

[37] M. Djerioui, Y. Brik, M. Ladjal, and B. Attallah, "Heart disease prediction using mlp and lstm models," in *2020 International Conference on Electrical Engineering (ICEE)*, pp. 1–5, IEEE, 2020. 2.2

[38] D. Chicco and G. Jurman, "Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone," *BMC Medical Informatics and Decision Making*, vol. 20, no. 1, p. 16, 2020. 2.3

[39] X. Song, X. Liu, F. Liu, and C. Wang, "Comparison of machine learning and logistic regression models in predicting acute kidney injury: A systematic review and meta-analysis," *International Journal of Medical Informatics*, vol. 151, p. 104484, 2021. 2.3

[40] T. Inoguchi, Y. Nohara, C. Nojiri, and N. Nakashima, "Association of serum bilirubin levels with risk of cancer development and total death," *Scientific Reports*, vol. 11, no. 1, p. 13224, 2021. 2.3

[41] S. Akter, H. U. Shekhar, and S. Akhteruzzaman, "Application of biochemical tests and machine learning techniques to diagnose and evaluate liver disease," *Advances in Bioscience and Biotechnology*, vol. 12, no. 6, pp. 154–172, 2021. 2.3

[42] D. W. Cockcroft and M. H. Gault, "Prediction of creatinine clearance from serum creatinine," in *Nephron*, vol. 16, pp. 31–41, Karger Publishers, 1976. 2.3

[43] A. Dauvin, C. Donado, P. Bachtiger, K.-C. Huang, C. M. Sauer, D. Ramazzotti, M. Bonvini, L. A. Celi, and M. J. Douglas, "Machine learning can accurately predict pre-admission baseline hemoglobin and creatinine in intensive care patients," *NPJ digital medicine*, vol. 2, no. 1, p. 116, 2019. 2.3

[44] W. Wang, G. Chakraborty, and B. Chakraborty, "Predicting the risk of chronic kidney disease (ckd) using machine learning algorithm," *Applied Sciences*, vol. 11, no. 1, p. 202, 2020. 2.3

[45] S. Kalasin, P. Sangnuang, and W. Surareungchai, "Lab-on-eyeglasses to monitor kidneys and strengthen vulnerable populations in pandemics: machine learning in predicting serum creatinine using tear creatinine," *Analytical Chemistry*, vol. 93, no. 30, pp. 10661–10671, 2021. 2.3

[46] E. Ghosh, L. Eshelman, S. Lanius, E. Schwager, K. S. Pasupathy, E. F. Barreto, and K. Kashani, "Estimation of baseline serum creatinine with machine learning," *American journal of nephrology*, vol. 52, no. 9, pp. 753–762, 2021. 2.3

[47] A. Aune, G. Vartdal, H. Bergseng, L. L. Randeberg, and E. Darj, "Bilirubin estimates from smartphone images of newborn infants' skin correlated highly to serum bilirubin levels," *Acta Paediatrica*, vol. 109, no. 12, pp. 2532–2538, 2020. 2.3

[48] I. Daunhawer, S. Kasser, G. Koch, L. Sieber, H. Cakal, J. Tütsch, M. Pfister, S. Wellmann, and J. E. Vogt, "Enhanced early prediction of clinically relevant neonatal hyperbilirubinemia with machine learning," *Pediatric research*, vol. 86, no. 1, pp. 122–127, 2019. 2.3

[49] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002. 3.1

[50] H. He, Y. Bai, E. A. Garcia, and S. Li, "Adasyn: Adaptive synthetic sampling approach for imbalanced learning," in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, pp. 1322–1328, IEEE, 2008. 3.1

[51] H. Han, W.-Y. Wang, and B.-H. Mao, "Borderline-smote: A new over-sampling method in imbalanced data sets learning," in *International Conference on Intelligent Computing*, pp. 878–887, Springer, 2005. 3.1

[52] A. J. Myles, R. N. Feudale, Y. Liu, N. A. Woody, and S. D. Brown, "An introduction to decision tree modeling," *Journal of Chemometrics: A Journal of the Chemometrics Society*, vol. 18, no. 6, pp. 275–285, 2004. 3.2

[53] G. Biau and E. Scornet, "A random forest guided tour," *Test*, vol. 25, pp. 197–227, 2016. 3.2

[54] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001. 3.2

[55] Z.-H. Zhou and J. Feng, "Deep forest," *National science review*, vol. 6, no. 1, pp. 74–86, 2019. 3.4

[56] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Machine Learning*, vol. 63, pp. 3–42, 2006. 3.4

[57] J. H. Friedman, T. Hastie, and R. Tibshirani, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer, 2nd ed., 2009. 4.1.1

[58] T. Fawcett, "An introduction to roc analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006. 5.5

[59] S. B. Kotsiantis, I. Zaharakis, and P. Pintelas, "Supervised machine learning: A review of classification techniques," *Emerging Artificial Intelligence Applications in Computer Engineering*, vol. 160, no. 1, pp. 3–24, 2007. 5.5