Binary Classification of Blood Values

Bachelor Thesis
Faculty of Science, University of Bern

submitted by

Fabian Gribi

from Bern , Switzerland

Supervision:

PD Dr. Kaspar Riesen
Institute of Computer Science (INF)
University of Bern, Switzerland

Abstract

Due to their ability to process large datasets efficiently, machine learning algorithms are of interest in many areas of medicine. Multidimensional data, such as electronic patient records, contain large amounts of information. However, datasets are not inherently suited to machine learning tasks. This thesis seeks to establish the viability of a specific dataset, which has not been utilized for machine learning purposes thus far. As a proof of concept, one-vs.-rest classification of ten select medical diagnoses is performed with XGBoost. Of the ten diagnoses, the best performing yielded a F_1-score of 0.782. The average F_1-score was 0.385. Although the overall performance was relatively low, further analysis of the feature importance confirms the feasibility of machine learning with this particular dataset.

Acknowledgements

I would like to extend my sincere thanks to my supervisor, PD Dr. Kaspar Riesen, who helped me find my thesis topic and whose guidance was invaluable. Furthermore, I would like to thank Prof. Dr. Alexander Leichtle, Dr. Harald Witte, and Tobias Blatter for their help in accessing and understanding the Swiss BioRef dataset. Lastly, I would like to express my gratitude to my family for their continued support. Special thanks go to my sister for lending me her expertise for many of the medical aspects of the thesis.

Contents

Chapter 1

Introduction

In this chapter, the research topic is introduced. Section 1.1 serves to familiarize the reader with the research context of the thesis. Subsequently, in Section 1.2, the research objective of the thesis is established. Finally, Section 1.3 outlines the structure of the remaining thesis chapters.

1.1 Topic Introduction

Machine learning is a branch of Artificial Intelligence concerned with using data to automatically derive rules, with which an output, usually a prediction or decision, can be generated. By learning the underlying patterns and structures of the data, machine learning algorithms can be applied to similar data or used to generate novel output. Machine learning algorithms have been used in many scientific disciplines outside computer science, such as economics, genomics, ecology, and medicine. Research into using machine learning has been ongoing in the medical field for decades [1]. Medical data, such as patient records, are now increasingly available in digital form. This is consequently accompanied by a growing number of published research papers on applying machine learning algorithms for processing these data [2]. Within medicine, machine learning algorithms can be used for various purposes such as research, prognosis, diagnosis, treatment, clinician workflow, and expanding the availability of clinical expertise [3][4].

Diagnostic applications have been investigated in many branches of medicine [5] for a large range of diseases. Hence, automated diagnosis is not restricted to a single type of data. Some applications use genetic information [6], others use image data [7]. Many applications use multidimensional data that combine patient information from different clinical tests, medications, etc. Much of the past research is focused on diagnosing a single disease or several diseases within a specific branch of medicine. Machine learning algorithms have been used to diagnose a variety of diseases, among which are chronic kidney disease [8], diabetes mellitus [9], and anemia [10]. Since hospital databases contain data that feature diagnoses from all branches of medicine, research where the focus is not placed on a single branch of medicine has also been conducted [11].

1.2 Research Goal

The work presented in this thesis is primarily exploratory. A partial dataset from a previous research project entitled Swiss BioRef [12] is utilized for the first time for machine learning purposes. The dataset encompasses measurements from 39 standard laboratory tests, aggregating data from 186,265 patients. Patient information is included with every measurement, consisting of the patient's age, gender, and relevant diagnoses. Diagnoses and laboratory tests are classified in a unified way following common taxonomies. With the amount of data and the valuable information contained within, finding novel uses for the dataset is worth pursuing.

Machine learning methods are suitable for processing large amounts of data such as the dataset mentioned above. Therefore, this thesis aims to assess the viability of the dataset for machine learning and identify future avenues of research. To do so, a specific application of machine learning is tested and evaluated. Namely, one-vs.-rest multi-class classification is performed for a subset of ten diagnoses. The machine learning model used is XGBoost (eXtreme Gradient Boosting) [13]. As a secondary goal of the work, factors that impact the model's performance, such as class imbalance, are investigated as well.

1.3 Thesis Overview

Here, the contents of the remaining chapters are outlined in brief. Chapter 2 reviews relevant concepts. This includes a discussion of machine learning in medicine. The chapter also reviews XGBoost and motivates its use. Chapter 3 describes how the dataset was prepared and discusses the specifics of the practical implementation. Following that, Chapter 4 discusses the results, which will mainly be an evaluation of the classification task. Finally, Chapter 5 closes the thesis with a conclusion on the work and an outlook on future research.

Chapter 2

Basics

In this chapter the theoretical background for later chapters will be presented. The first section is concerned with machine learning in medicine. It will introduce relevant machine learning concepts and give an overview of machine learning applications in medicine. The subsequent section motivates using XGBoost as the sole machine learning model used in this thesis and discusses essential aspects of the model.

2.1 Machine Learning in Medicine

2.1.1 Machine Learning Terminology

Machine learning algorithms are usually divided into categories based on the specific task they perform. Tasks are defined by the data used to train the model and the desired output. In the field of machine learning, three paradigms have become the standard for classifying algorithms. These are:

- *Supervised learning:* Supervised learning refers to machine learning tasks where labeled data is used, meaning the input data and their corresponding correct outputs are provided during training. The aim is to derive a mapping from input to output that can be applied to unseen data. The two most common types of supervised learning tasks are classification (prediction of discrete values) and regression (prediction of continuous values).

- *Unsupervised learning:* In unsupervised learning, the model is presented with unlabeled data during training. The goal consists of discovering patterns, relationships, and structures within the data without specific guidance from a human. Clustering and dimensionality reduction are common examples of unsupervised learning tasks. Clustering involves grouping similar data points, while dimensionality reduction aims to reduce input features while preserving the data's essential information.

- *Reinforcement learning:* Reinforcement learning refers to a form of machine learning where the model emulates an intelligent actor interacting with its environment. The goal is to find the optimal action based on the environment's current state. This type of machine learning is therefore well suited for robotics and game playing.

This thesis is concerned with classification and therefore falls within the supervised learning paradigm.

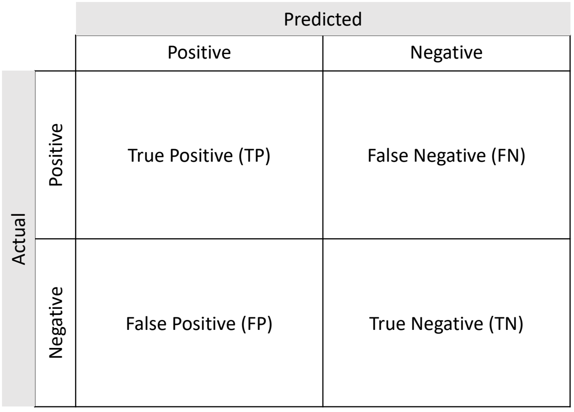
Bias-Variance Tradeoff

A well-known obstacle in supervised learning is the bias-variance tradeoff. In the application of supervised models, good performance on the training data should also translate into good performance on unseen data. This is where bias and variance enter a reciprocal relationship. Bias refers to errors introduced by the model not being sensitive enough to relationships between input data and the output labels. Variance, in turn, refers to errors introduced due to the model being overly sensitive to slight variations within the training data. High variance models perform well on training data but do not generalize to unseen data. This is referred to as overfitting. High-bias models perform poorly both on unseen data and on training data. For optimal performance data, bias and variance have to be balanced such that a model recognizes patterns but does not learn minor variations present in the training data. Specific measures to combat overfitting related to the machine learning model used in this thesis are discussed in Section 2.2.1.

Evaluation Metrics

Classification tasks are referred to as such because the discrete outputs can be conceptualized as classes, and the outputs often represent distinct concepts in practical applications. Binary classification refers to tasks with only two possible classes. In many cases, the classes are referred to as positive class and negative class, or simply as "1" and "0".

Evaluating a model's performance with only accuracy, meaning the fraction or percentage of correctly predicted data, often is inadequate since it neglects to evaluate performance on individual classes. In binary classification there are several metrics to assess a model's performance more comprehensively. One standard method to visualize performance is to use a confusion matrix. Figure 2.1 shows an example



		Predicted	
		Positive	Negative
Actual	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

Figure 2.1: **Confusion matrix**

of how a confusion matrix is structured. The number of correctly identified data are shown in their respective square for both negative and positive classes. Other metrics can be derived as functions of these fundamental measurements. Relevant to the evaluation in Chapter 4 are precision, recall, and the F_1-score.

$$\begin{aligned} Precision &= \frac{TP}{TP+FP} && (2.1) \\ Recall &= \frac{TP}{TP+FN} && (2.2) \\ F_1\text{–}score &= 2 \times \frac{Precision \times Recall}{Precision + Recall} && (2.3) \end{aligned}$$

Precision indicates the fraction of actually positive data points among the entirety of data predicted to be positive. Recall, or sensitivity, is the fraction of positive data that were correctly labeled as such. These two metrics are not independent, and an increase in one might lead to a decrease in the other. For example, an increase in recall caused by a larger number of positive predictions might lead to a reduction in precision due to more false positives. The F_1-score is the harmonic mean of precision and recall. It was chosen as the principal evaluation metric for the experimental part since it represents both earlier metrics as a single value.

2.1.2 Applications in Medicine

As in other research fields, the interest in finding machine learning applications has increased since the turn of the century. Shehab *et al.* conducted a literature review of over 200 publications from 2000 to 2022 [2]. They note a clear upward trend in the use of the term ”machine learning in the medical field”. In their work, publications are grouped primarily according to specific diseases. A broader analysis of machine learning applications in the medical field is provided by Rajkomar, Dean, and Kohane [3]. The following is a list of the uses they identify:

- *Prognosis:* Leveraging large amounts of patient data, machine learning algorithms might be able to predict specific health outcomes better than a single clinician. One example given for an application already in use is a system that identifies patients at risk for needing intensive care in the future [14].
- *Diagnosis:* Mistakes in diagnosis are fairly common. A 2015 report by the National Academies of Sciences, Engineering, and Medicine found that most patients experience at least one diagnostic error in their lifetime [15]. Machine learning algorithms that assist clinicians might lead to a reduction of errors.
- *Treatment:* For many patients, multiple options are available in treatment. By using data from former patients, treatment can be adapted to a specific patient for optimal outcomes.
- *Clinician workflow:* Practicing clinicians treat several patients per day and are burdened with extracting relevant patient information from their records. Using techniques such as automatic summarization or voice dictation, clinicians could spend less time reading and writing patient records and focus on treatment.
- *Expanding the availability of clinical expertise:* Another challenge to providing healthcare is that clinicians are unlikely to be able to interact with all the patients in need of care physically. Automated triage could help identify patients who require care the most, thus decreasing demand for clinicians and improving availability.

It is crucial to emphasize that these applications never fully replace a trained clinician and only serve as helpful tools that either extend their capabilities or replace a specific part of their work. Another thing to note is that this list focuses exclusively on applications in clinical practice. Other possibilities exist to utilize machine learning in medicine outside of clinical practice, such as the discovery and development of novel drugs [4]. In this thesis, patient data are classified according to certain diseases, meaning that the topic would fall within the paradigm of diagnosis.

Machine Learning for Diagnostic Purposes

Researchers have applied machine learning methods to diagnose various diseases. Typically, these approaches are focused on a single disease. Certain medical disciplines receive more attention than others. In a 2021 review, Bhavsar *et al.* observe that the two most common disciplines where machine learning diagnosis is used are

cardiology and endocrinology [5]. They identify the prevalence of the associated diseases and the resulting availability of data as the main reason for this observation. Other disciplines where machine learning has been used for diagnostic purposes include oncology [6], nephrology [8] and pulmonology [16].
A critical aspect of machine learning applications in diagnostics is the interpretability of the model. Black-box models, where there is little mechanistic interpretability, possibly hurt the trust in the model and prevent disclosure of important information to patients [17]. It is crucial to note that simpler models, such as linear ones, are not inherently more interpretable than more complex models, e.g., neural networks. In certain aspects, deep-learning models might actually be more interpretable, since deep-learning models typically do not require extensive feature engineering and often offer good post hoc interpretability since they learn rich representations that can, for example, be visualized [18].

Related Work

The approach of using a dataset with patient data to predict diagnoses that are not from the same medical discipline is uncommon. In the following paragraphs, the focus will therefore be on presenting examples where binary classification is applied to specific diseases that are also used in this thesis. Finally, one publication with a similar approach to this thesis is reviewed.
Ogunleye and Wang apply binary classification for the diagnosis of chronic kidney disease [8]. For model selection, they compare the base performance of several machine learning models: Linear discriminant analysis, classification and regression trees, support vector machines, k-nearest neighbor, and XGBoost. As XGBoost achieved the best base performance, it was elected as the model for the remainder of their work. After optimizing and training, they report an accuracy of 1.000, a sensitivity of 1.000, and a specificity of 1.000. Furthermore, a reduced model was constructed based on the most relevant features. For the reduced model, they report identical scores as for the complete model.
Ismail *et al.* evaluate 35 different machine learning algorithms in the binary classification of type 2 diabetes on three different datasets [9]. By using consistent evaluation metrics, they intend to obtain results to compare the models objectively. They report high accuracy for all models on all three datasets. But for many models, the reported F_1-score is zero since none of the samples belonging to the positive class were correctly predicted. They attribute this to high class imbalance within the datasets used, ultimately hindering the reported goal of comparing models objectively.
Azarkhish, Raoufy, and Gharibzadeh use both an artificial neural network (ANN)

and an adaptive neuro-fuzzy inference system (ANFIS) to diagnose iron deficiency anemia using results from common laboratory blood tests [10]. Furthermore, they predict serum iron levels. For the evaluation of the predicted diagnoses, the two models are evaluated against each other and against predictions obtained by logistic regression. The reported accuracy is 0.9629 for the ANN, 0.9074 for the ANFIS, and 0.6296 for logistic regression. From the reported precision and recall, the following F_1-scores can be calculated: 0.9675 for the ANN, 0.9151 for the ANFIS, and 0.7434 for logistic regression.

The methods that are most similar to this thesis' approach can be found with Sakhibgareeva and Zaozersky [11]. They use a dataset containing established diagnoses, laboratory test results, and patient information such as age and gender. They restrict prediction to four diagnoses[1]: iron-deficiency anemia (D50), non-insulin-dependent diabetes mellitus (E11), other disorders of carbohydrate metabolism (E74), and disorders of lipoprotein metabolism and other lipidemias (E78). Using gradient boosting of decision trees, they perform classification in a one-vs.-rest approach, reducing it to four binary classification tasks. The reported accuracy values are 0.95 for D50, 0.90 for E11, 0.97 for E74, and 0.89 for E78. Similar to the previous publication, the F_1-score is not explicitly stated but can be calculated from the reported precision and recall values, resulting in the following scores: 0.7353 for D50, 0.6531 for E11, 0.3111 for E74, and 0.9237 for E78.

2.2 Selected Machine Learning Algorithm

This section introduces theoretical concepts related to XGBoost that are relevant to later chapters and motivates the use of XGBoost.

2.2.1 Details about the Algorithm

XGBoost (extreme gradient boosting) was introduced by Chen and Guestrin [13]. The XGBoost library supports multiple types of tasks and multiple models. The discussion here will be limited to the tree-based gradient boosting model. XGBoost differs in many regards from other gradient boosting systems. Most differences are in the technical implementation, not the underlying algorithm. However, XGBoost diverges from other algorithms in certain aspects, such as its split-finding mechanism and the use of a regularized objective. In this section, the focus will lie on reviewing the fundamentals of the underlying algorithm, but the regularized objective will also be discussed briefly.

[1] Corresponding ICD-10-GM codes in parentheses

Decision Trees

The fundamental building block of XGBoost is the decision tree. Decision trees are machine learning algorithms that split an input space $\mathcal{D} = \{(x_i, y_i)\}$ $(x_i \in \mathbb{R}^m, y_i \in \mathbb{R})$ into disjoint subsets. The algorithm starts by splitting the original set $\mathcal{D}$ into two subsets and then recursively splitting the subsets. Splits are made based on thresholds for single features. More formally, a parent set $\mathcal{D}_p$ is split into two subsets $\mathcal{D}_1, \mathcal{D}_2$ based on a given threshold $\theta \in \mathbb{R}$. These child subsets are defined as

$$\mathcal{D}_1 = \{x_i \mid x_{ij} < \theta,\ x_i \in \mathcal{D}_p\}$$

$$\mathcal{D}_2 = \{x_i \mid x_{ij} \geq \theta,\ x_i \in \mathcal{D}_p\}$$

Where $j \in \{0, \ldots, m-1\}$ is the index of the feature to which the threshold is applied.

Splits are made in a manner that maximizes a given loss function, e.g., mean squared error. The splitting procedure is repeated until certain stopping criteria are met, for example, a fixed number of iterations. The subsets created in the last iteration of splitting are referred to as leaves. Depending on the task, predictions are made by either taking the majority class at every leaf (classification) or the mean value of the leaf (regression). Decision trees can be written in an additive form. Let T be the number of leaves, $\mathcal{D}_k$ the disjoint leaf sets, and $w \in \mathbb{R}^T$ the vector containing the prediction values of the leaves. Then the tree can be written as:

$$f(x) = \sum_{k=1}^{T} w_k \mathbb{1}_{\mathcal{D}_k}(x) \tag{2.4}$$

Where $\mathbb{1}_{\mathcal{D}_k}(x)$ is defined as

$$\mathbb{1}_{\mathcal{D}_k}(x) = \begin{cases} 1 & \text{if } x \in \mathcal{D}_k \\ 0 & \text{else} \end{cases}$$

There are several decision tree algorithms. The one employed by Chen and Guestrin for XGBoost is CART (classification and regression trees) introduced by Breiman *et al.* [19].

Gradient Boosting

Boosting is a method by which weak learners, such as decision trees that only make a few splits, are combined to make predictions. Gradient boosting, more specifically TreeBoost proposed by Friedman [20], employs forward stagewise additive modeling, meaning trees are added iteratively to an initial prediction during training. Each new tree is constructed to correct the errors of the previous prediction. At iteration t the prediction $\hat{y_i}^t$ for input x_i is given by

$$\hat{y_i}^t = \hat{y}_i^{t-1} + f_t(x_i) \tag{2.5}$$

Where $\hat{y}_i^{t-1}$ is the prediction from the previous iteration and f_t is the new tree to be added. Given some loss function l this prediction is optimized by minimizing the following objective function:

$$\mathcal{L}^{(t)} = \sum_{i=0}^{n} l(y_i, \hat{y}_i^{t-1} + f_t(x_i)) \tag{2.6}$$

Where y_i is the actual value that corresponds to the input x_i. Friedman proposes solving this optimization problem by using the second-order approximation of $\mathcal{L}^{(t)}$ [20]. From the minimum, the optimal leaf values for f_t can be derived, which are used for the split-making process to approximate the optimal structure for f_t.

Regularization

The method proposed by Chen and Guestrin follows Friedman's gradient boosting implementation but expands the objective function by also including a regularization term Ω, resulting in the following objective function:

$$\mathcal{L}^{(t)} = \sum_{i=0}^{n} l(y_i, \hat{y}_i^{t-1} + f_t(x_i)) + \Omega(f_t) \tag{2.7}$$

They define Ω as:

$$\Omega(f) := \gamma T + \frac{1}{2}\lambda \|w\|_2^2 \tag{2.8}$$

Where T is the number of leaves that f has and w is the vector of leaf weights of f. Both γ and λ are hyperparameters that can be adjusted. Higher values for γ and λ increase the objective function. If carefully chosen, these parameters help prevent overfitting by punishing complex tree structures.

2.2.2 Motivation for Using XGBoost

In any classification task, the question arises as to which model should be used. XGBoost was selected for several reasons. The primary reason is its performance, both with regard to classification and computational efficiency. XGBoost has been shown to outperform simpler algorithms, such as decision trees or support vector machines, in classification [8] tasks. In addition, it has been demonstrated that XGBoost is significantly faster than other implementations of gradient boosting. XGBoost was reported to be ten times faster when learning compared to scikit-learn's implementation of gradient boosting [13]. Furthermore, the XGBoost library is open source[2]. Lastly, XGBoost, despite its structural complexity, offers post hoc interpretability in that the library provides methods to analyze feature importance. This is especially relevant for machine learning tasks in the medical field, as discussed in Section 2.1.2.

[2] Code available at https://github.com/dmlc/xgboost

Chapter 3

Methods

The upcoming chapter will provide an introduction to the methods employed. The chapter will begin with a discussion of the dataset. This will include information about its origin, its structure, and how it was processed for the experimental part. The hyperparameter tuning procedure will be explained in the latter section of the chapter.

3.1 Dataset

3.1.1 Swiss BioRef

The dataset used for this thesis is part of a larger dataset created for a research project entitled Swiss BioRef[1] [12]. The stated aim of the project is to create a sustainable framework for assessing patient-group-specific reference intervals. To do so, laboratory data from four major Swiss hospitals were collected. The fraction used in this thesis is the data collected from the University Hospital Bern.

3.1.2 Structure of the Dataset

The data consist of 5,967,846 anonymized laboratory blood measurements from 186,265 patients. Four attributes compromise a measurement: a description of the laboratory test, a code identifying the laboratory test, a value, and a unit of measurement. The codes follow the Logical Observation Identifiers Names and Codes (LOINC) standard [22]. The set contains measurements from 39 different tests. A complete list of the laboratory tests is available in Appendix A. In addition to the measurement, each entry contains information about the patient. This includes the

[1] The paper about the project itself has not yet been published. The citation listed is a preprint. However, specific technical aspects have been published (see Fasquelle-Lopez and Raisaro [21])

age, administrative gender, and up to five previously established diagnoses considered relevant. Diagnoses are encoded according to the ICD-10-GM taxonomy[2] [23]. Lastly, there is information about the laboratory device used for the test. In total, a single entry consists of 21 features. A complete list of the features is available in Appendix A.

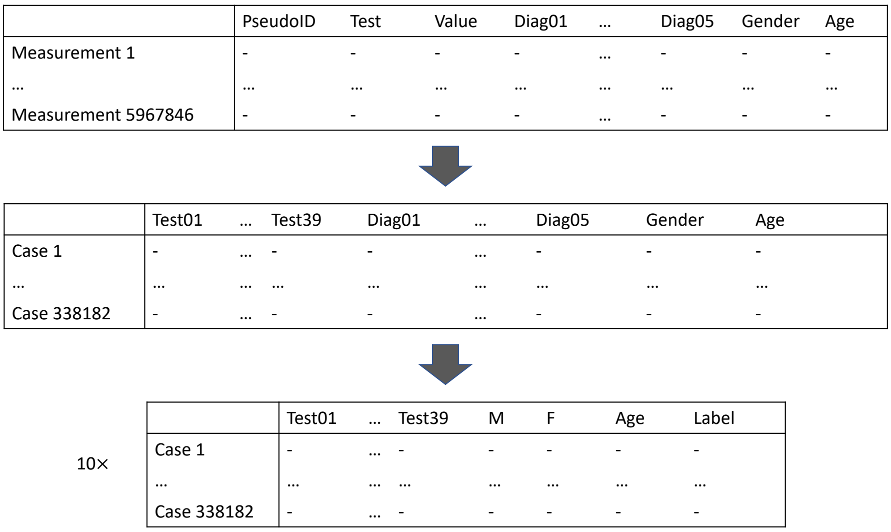
One caveat worth noting about the diagnoses is that they precede the laboratory tests. Since the data were collected to assess reference values, the inclusion of diagnoses aims to provide information about the patient's belonging to a specific patient group, e.g., people with type 2 diabetes. Diagnoses were therefore not made as a consequence of the laboratory results but rather function as metadata about the measurements. This is a significant limitation to conducting the main part of this thesis since a statistical relationship between predictor variables, i.e., the laboratory results, and diagnosis is assumed for classification tasks.

Another substantial issue arising from the fact that the dataset was created to assess reference values also requires clarification. The dataset is grouped by measurement, meaning multiple entries may belong to the same patient; this is often the case, as the average patient has around 32 entries. Multiple measurements may be relevant for a certain diagnosis, and using all measurements as predictors for a single patient is desirable for the classification task. Since the data are anonymized, there exists no reliable way of grouping the entries by individual patients without deanonymizing the data. The method to circumnvent this problem is discussed in the next section.

3.1.3 Preprocessing

As discussed in the previous section, grouping measurements by the patient is impossible. However, the data can be restructured to give an approximation of the cases. Cases differ from patients in that a patient can have multiple cases. To construct the approximate cases, measurements with the same patient information, meaning age, gender, and diagnoses, that are directly consecutive are assumed to belong to the same case and are thus grouped together. The measurements can then be used as features for a single entry. This restructuring is illustrated as the first step in Figure 3.1. For simplicity, the representation of the unchanged dataset at the top of the figure only shows a subset of the actual features. Features with information about laboratory equipment were disregarded during this first step and the remainder of the thesis, with the assumption that variation introduced by dif-

[2] ICD-10-GM does not only encode diagnoses of diseases but also other patient statuses. In the remainder of the thesis "diagnosis" is therefore not used in its more colloquial sense, but rather it is used to mean patient condition codified in the ICD-10-GM texonomy.



	PseudoID	Test	Value	Diag01	…	Diag05	Gender	Age
Measurement 1	-	-	-	-	…	-	-	-
…	…	…	…	…	…	…	…	…
Measurement 5967846	-	-	-	-	…	-	-	-

	Test01	…	Test39	Diag01	…	Diag05	Gender	Age
Case 1	-	…	-	-	…	-	-	-
…	…	…	…	…	…	…	…	…
Case 338182	-	…	-	-	…	-	-	-

10×

	Test01	…	Test39	M	F	Age	Label
Case 1	-	…	-	-	-	-	-
…	…	…	…	…	…	…	…
Case 338182	-	…	-	-	-	-	-

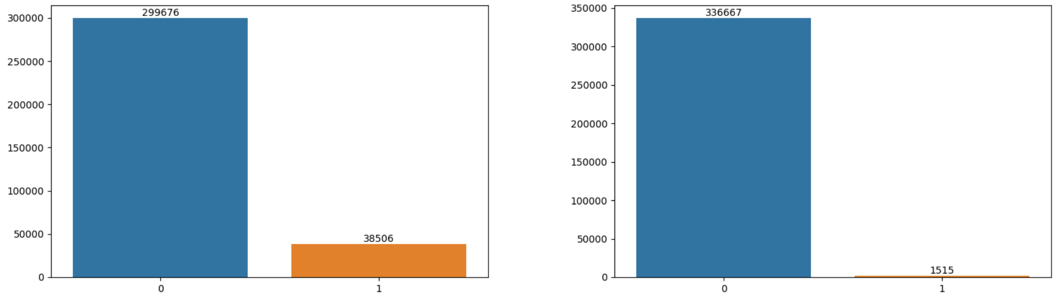
Figure 3.1: **Schematic representation of dataset preparation.**

ferent laboratory equipment would not significantly impact the classification.

The dataset contains measurements from 323,600 cases. After reshaping the data as described above, the number of cases is 338,182. The method is therefore not optimal, but it suffices as a conservative approximation. Notwithstanding, this should be considered a limitation to the significance of the results that will follow in Chapter 3.

After this initial reshaping of the data, copies of the dataset were created for each target diagnosis. The total number of different diagnoses exceeded 1600. For practical reasons, the classification was therefore limited to ten diagnoses. Binary labels were assigned, indicating for each constructed case whether the target diagnosis was listed. Cases where the diagnosis was present were assigned to the positive class and cases where the diagnosis was absent were grouped in the negative class. For an overview of the distribution of the laboratory results of the two classes, letter-value plots are provided in Appendix B. Lastly, one-hot encoding was used for the gender feature to avoid having categorical features.

Diagnoses were selected for different reasons. Table 3.1 gives an overview of the diagnoses and why they were chosen. The four most frequent diagnoses, i.e., the diagnoses listed the most after the approximation of the cases, were chosen. Three diagnoses were included due to their potential to work well with the given classification task. Both diabetes and anemia affect certain blood values. Diabetes is also a popular subject of machine learning research and has been studied previously (e.g., Ismail *et al.* [9]). Specifically, type 2 diabetes and iron deficiency anemia were selected because both are the most common form of the overarching diagnosis in



(a) I25

(b) F43

Figure 3.2: **Class imbalance for the most and least common diagnosis.** Number of cases with (1) and without (0) diagnosis shown for chronic ischemic heart disease (I25) and reaction to severe stress and adjustment disorders (F43)

the dataset. Duration of pregnancy is also included. Since duration is not specified further in the data, O09 will be referred to as pregnancy. Pregnancy was chosen in response to the limitation that the diagnoses precede the laboratory tests. Pregnancy is unlikely to be listed as a diagnosis if the person was not pregnant at the time of blood sample collection. Thus, possible changes in the measured blood values would more likely manifest than for other diagnoses. Lastly, the three least common diagnoses are chosen, with the added condition that there must be more than 1,500 cases for the diagnosis to be included. Many diagnoses are listed only once or twice in the entire data. Setting a lower bound was deemed necessary since classification with such few positive samples would not be practicable.

The intention of including diagnoses with a higher negative-to-positive class ratio is to see how well the classifier fares with imbalanced data. Class imbalance may refer to any dataset where class labels are imbalanced, but the common understanding is that class imbalance refers to more significant differences. There is no precise definition of what ratio is sufficient to constitute a class imbalance, but there are heuristics. He and Garcia observe that more severe class imbalance, referred to as *between-class imbalance*, commonly includes ratios of 100:1 to 10,000:1 [24]. Hence, 1,500 was elected as a lower limit since the resulting ratio falls within the informal definition of *between-class imbalance* while still having an adequate number of positively labeled data.

Even with the most common diagnosis, the ratio between negative and positive class is almost 9:1. Figure 3.2 shows a bar plot with the number of cases with and without the diagnosis for the most common and least common diagnosis. Bar plots for the remaining diagnoses are provided in Appendix B.

Reason	Code	Description	Number of Cases
Most Common	I25	Chronic ischemic heart disease	38506
	I10	Essential (primary) hypertension	38440
	N18	Chronic kidney disease	24478
	I50	Heart failure	19053
Classification Potential	E11	Diabetes mellitus, type 2	15983
	O09	Duration of pregnancy	12019
	D50	Iron deficiency anemia	5102
Least Common (n >1500)	P39	Other infections specific to the perinatal period	1550
	A46	Erysipelas	1518
	F43	Reaction to severe stress and adjustment disorders	1515

Table 3.1: **Chosen target diagnoses.** The codes follow ICD-10-GM. Corresponding definitions are listed in the description column. For each diagnosis, the number of positively labeled data is indicated in the rightmost column.

3.2 Hyperparameter Tuning

Hyperparameters are parameters that influence how the model learns from the data and are not learned themselves. The choice of hyperparameters can significantly impact a model's performance. Hyperparameter tuning is therefore a critical aspect of optimizing machine learning models. For binary classification with a tree-based model, XGBoost provides numerous parameters that can be adapted. To avoid unnecessary complexity, tuning of parameters was therefore restricted further to the eight parameters shown in Table 3.2.

These hyperparameters control different aspects of the model. Both reg_lambda and gamma are regularization parameters that penalize complex models by increasing the function value in Equation 2.8. Gamma can also be interpreted as the minimum loss required to make a specific split. Another regularization term not included in the function from the paper is reg_alpha, which is an L1 regularization term, meaning that it scales the penalty from having more extreme values on the leaves. Higher values for these three parameters leads to a more conservative model, i.e., a model less prone to overfitting. The learning_rate parameter is a shrinkage factor scaling the correction from added trees at boosting iterations. Hence, a lower value for learning_rate tends to favor a more conservative model. The parameter colsample_bytree governs column sampling. It indicates the fraction of features used for constructing the trees. For any value below one only a subset of the features is used for constructing an individual tree. Max_depth controls the depth of the trees and min_child_weight controls the minimum sum of instance weights needed in a child node. To address class imbalance, scale_pos_weight can be used since it scales the gradients for positive samples, making them more influential. Since the positive

Hyperparameter	Description	Range
learning_rate	Shrinkage factor for boosting steps. Lower values decrease the correction from new trees.	$[0, 1]$
scale_pos_weight	Scaling factor for positive class. Higher values favor correcting errors in the positive class.	$(0, \infty)$
gamma	Minimum loss reduction required to make a split.	$[0, \infty)$
max_depth	Maximal depth for trees.	$\mathbb{N}_{>0}$
min_child_weight	Minimum sum of instance weight needed in a child node. Larger values lead to less partitioning.	$[0, \infty)$
colsample_bytree	Subsample of features used for constructing each tree. Lower values lead to a more conservative model	$(0, 1]$
reg_alpha	L1 regularization term.	$[0, \infty)$
reg_lambda	L2 regularization term.	$[0, \infty)$

Table 3.2: **Hyperparameters selected for tuning.** The rightmost column indicates theoretical ranges of possible values.

class is also the minority class, an appropriate increase for this parameter can create a better balance between the majority and minority class.

Different approaches can be taken to tune hyperparameters. One method is to employ a grid search, in which predetermined parameter ranges are searched by evaluating performance for all possible parameter combinations. This method has the obvious downside that complexity increases rapidly with the number of hyperparameters. Another way is to select parameter values from a range according to some probability distribution. Models tuned with random search have been shown to perform as well as models where grid search was applied [25] with the benefit that random search is less computationally expensive. Both grid and random search do not incorporate information from previous iterations when selecting new parameter combinations. Bayesian optimization on the other hand does so, and thus more efficiently searches the parameter space. Furthermore, Bayesian optimization has been used in combination with XGBoost in the past [26]. Therefore Bayesian optimization, implemented in the Python library Hyperopt [27], was used as the method of choice in this thesis. In addition to Hyperopt, scikit-learn [28] was used in the tuning process for performing the cross-validation.

The complete procedure for tuning the hyperparameters was repeated for each of the ten datasets that were created as described in Section 3.1.3 in order to create a separate model for each diagnosis. First, a stratified split was applied to use 80% of the data as a training set and the remaining 20% as a test set. Instead of a validation set, stratified 10-fold cross-validation of the training dataset was used. The function to be maximized by Bayesian optimization was the average F_1-score on the ten subsamples created at every iteration. The maximum number of iterations for the optimization was set as 50. After termination, the optimized parameters were used to fit the model on the training data. The model is then evaluated on the test dataset.

The F_1-score is used as a target evaluation metric in the optimization since it creates a better balance between the classification of the majority and minority class. If, for example, accuracy were to be used, the choice of parameters would favor correctly classifying the majority class and neglecting the minority class. Therefore, using the F_1-score is another way of addressing class imbalance.

Chapter 4

Experimental Evaluation

This chapter demonstrates and discusses the classifiers' performance. In the first section, their performance will be evaluated against two different baselines. In addition, there will be a discussion of the difference in performance between the training dataset and the test dataset. The first section will conclude with an examination of the feature importance. In a second section, select results will be examined further. This will include exploring different target evaluation metrics in the hyperparameter selection.

4.1 Model Evaluation

4.1.1 Comparison to Baseline

As seen in section 3.1.3 the datasets for the different diagnoses are all somewhat imbalanced, with most of the data points being labelled negative. By simply labelling all data points as the majority class, a naïve classifier could already perform well with respect to certain evaluation metrics. An actual classifier trained on the data should outperform this naïve classifier. Hence, in Table 4.1 the performance of the model and two different baselines are shown.

In the table, zero rate classifier refers to the method described above where all data points are assigned the negative label. Note that the precision is zero for this classifier for all diagnoses. Precision cannot be defined in this case since no samples are assigned the positive class leading to a division by zero in the calculation of the score (see Equation 2.1). In this case, precision is set to zero as a default. For the random rate classifier labels are generated randomly. Class distribution is considered by generating the labels according to the class imbalance of the dataset. For both of these baseline classifiers and the actual classifier accuracy, precision, recall and the F_1-score are shown across the diagnoses. Diagnoses are indicated with their

Diagnosis	Zero Rate Classifier				Random Rate Classifier				Trained Classifier			
	Acc	Prec	Rec	F_1	Acc	Prec	Rec	F_1	Acc	Prec	Rec	F1
I25	0.886	0.000	0.000	0.000	0.796	0.113	0.116	0.114	0.917	0.669	0.533	0.593
I10	0.886	0.000	0.000	0.000	0.799	0.118	0.119	0.119	0.775	0.275	0.601	0.377
N18	0.928	0.000	0.000	0.000	0.865	0.073	0.074	0.073	0.900	0.384	0.638	0.480
I50	0.944	0.000	0.000	0.000	0.894	0.053	0.052	0.052	0.915	0.308	0.403	0.349
E11	0.953	0.000	0.000	0.000	0.910	0.044	0.043	0.043	0.920	0.298	0.520	0.379
O09	0.964	0.000	0.000	0.000	0.933	0.035	0.034	0.035	0.982	0.696	0.894	0.782
D50	0.985	0.000	0.000	0.000	0.970	0.008	0.009	0.009	0.977	0.304	0.405	0.347
P39	0.995	0.000	0.000	0.000	0.990	0.003	0.003	0.003	0.994	0.337	0.361	0.349
A46	0.996	0.000	0.000	0.000	0.991	0.003	0.003	0.003	0.990	0.107	0.171	0.132
F43	0.996	0.000	0.000	0.000	0.991	0.007	0.007	0.007	0.986	0.046	0.109	0.065

Table 4.1: **Model performance in comparison to two baselines.** Accuracy (Acc), precision (Prec), recall (Rec) and F_1-score (F_1) for zero rate classifier, random rate classifier, and actual predictions on test dataset. For the predictions on the test dataset the highest value for each of the metrics is indicated in red.

ICD-10-GM code. Descriptions are provided in Table 3.1

Baseline accuracy is high since it relates to the fraction of negative labels in the datasets. For the zero rate classifier recall is zero for all diagnoses since none of the samples from the positive class were labelled as such. The random rate classifier performs better in that respect since a few positive labels are generated correctly. Precision is also better since there are a few true positives. Therefore precision can be calculated, resulting in a higher score than the default score for the zero rate classifier.

The accuracy of the actual predicted values decreases from the baseline in all but two cases. The exceptions are chronic ischemic heart disease (I25) and pregnancy (O09), where predictions are more accurate than the baseline. The reasons for the lower relative accuracy are twofold: first, the accuracy for the baselines is high due to class imbalance, and second the F_1-score is used as an evaluation metric for parameter selection. The F_1-score weighs the minority class more significantly than the accuracy score, improving performance in that respect over the baseline. In return, the misclassification rate in the majority class might increase leading to an overall lower accuracy.

The classifiers outperform the baseline precision and recall values in all diagnoses. Consequently, this also applies to the F_1-score. The F_1-score is nonetheless quite low and would likely be insufficient for actual clinical use. Performance requirements across regulatory bodies are inconsistent and usually depend on the area of application. For example, a higher false positive rate for pre-screening diseases is less problematic. On the other hand, false negatives should be avoided as much as possible in such a use case. Since no specific target was set for this thesis the performance cannot be evaluated objectively. However, compared to the reported

performance values for the same diagnoses in Section 2.1.2 the models from this thesis all yield a worse F_1-score.
Poor performance can have several underlying causes and these causes are not necessarily the same for all ten classifiers. The following is a non-exhaustive list:

- *Little to no correlation between the diagnosis and the features:* Most of the features are laboratory blood test results. Some diagnoses might not impact any of these values.

- *Diagnoses preceding measurements:* This limitation was introduced in section 3.1.2. Decreased correlation between target and predictor variables leads to bad separability of the classes.

- *Class imbalance:* Imbalance within the training dataset might lead to the classifier skewing towards labeling the samples as negative.

- *Issues with the model:* Though XGBoost performs well on classification tasks (e.g. diagnosing kidney disease [8]) unsuitable choice of hyperparameters could lead to bad performance.

The first point can be disregarded as a cause with the diagnoses that were explicitly chosen for their symptoms manifesting in a change of specific blood values, i.e., type 2 diabetes (E11) and iron deficiency anemia (D50). Class imbalance also does not seem to be the main issue in the case of type 2 diabetes since the classifier for pregnancy outperforms the classifier for diabetes, despite the pregnancy dataset being more imbalanced. Hence the two most likely causes for the bad performance with these diagnoses are the parameter choice or a weak relationship between the diagnoses and the laboratory results.

4.1.2 Comparison to Training Dataset

Despite using cross-validation when performing the parameter search, overfitting might occur especially since there was no separate validation set used. Here, the model's performance on the training set is compared to performance on test data to give an impression of how well the models generalize. If the model performs significantly better on the training dataset than the test dataset, the model is likely overfitted to the training data. Table 4.2 shows the model's performance on both the test and training dataset. For each column, the highest value is indicated in red.
Comparing the two sets of performance evaluations, it is evident that there is a decrease in all metrics from the training dataset to the test dataset. The decrease is

Diagnosis	Training Dataset				Test Dataset			
	Acc	Prec	Rec	F_1	Acc	Prec	Rec	F_1
I25	0.951	0.828	0.723	0.772	0.917	0.669	0.533	0.593
I10	0.813	0.358	0.811	0.497	0.775	0.275	0.601	0.377
N18	0.914	0.442	0.737	0.552	0.900	0.384	0.638	0.480
I50	0.977	0.714	0.995	0.831	0.915	0.308	0.403	0.349
E11	0.936	0.402	0.744	0.522	0.920	0.298	0.520	0.379
O09	0.985	0.721	0.933	0.813	0.982	0.696	0.894	0.782
D50	0.984	0.488	0.718	0.581	0.977	0.304	0.405	0.347
P39	0.996	0.541	0.586	0.562	0.994	0.337	0.361	0.349
A46	0.991	0.220	0.427	0.290	0.990	0.107	0.171	0.132
F43	0.989	0.255	0.700	0.374	0.986	0.046	0.109	0.065

Table 4.2: **Performance on test dataset in comparison to performance on training dataset.** Accuracy (Acc), precision (Prec), recall (Rec), and F_1-score (F_1) for both datasets. The highest achieved value in each metric is indicated in red for both datasets.

not consistent across diagnoses. The most significant difference in the performance for all metrics occurs for heart failure (I50). For the least pronounced difference in the metrics, there is no single diagnosis. The smallest difference in accuracy is found with erysipelas (A46). For the remaining metrics, the least pronounced gap is found with pregnancy (O09).

This leads to the conclusion that the individual classifiers all show some level of overfitting. Though the difference for heart failure seems more excessive. This might simply be a consequence of the specific split into training and test dataset for that diagnosis. In subsection 4.2.1 this possibility will be revisited.

4.1.3 Feature Importance

An advantage of using XGBoost rather than a deep learning model is that the model retains some level of transparency and interpretability by offering different metrics on feature importance. By inspecting feature importance, one can check if the features that impact the classification most are features that relate to the diagnosis.

In this subsection, gain will be used as the sole metric to gauge feature importance. Gain is a measure of how much a single feature contributes to the overall accuracy of the classifier. Importance is calculated first for each tree in the ensemble by taking the amount a split on a specific feature improves the performance measure. The final gain for a specific feature is the average over these importance values from all trees.

Most features do not warrant inclusion in the discussion due to their low relative

Diagnosis	Most Important Features		
	1st	2nd	3rd
I25	HDL-C	f	Age
I10	Age	HDL-C	CH
N18	GFR	C	Age
I50	Age	HDL-C	U
E11	G	HBA1C	Age
O09	Age	PT	m
D50	HCT	MCH	Age
P39	Age	CRP	GFR
A46	CRP	O	CHL
F43	Age	CRP	GFR

Table 4.3: **The three most important features for each diagnosis according to gain.** Since one-hot encoding is used for gender both male (m) and female (f) are present as separate features. The featured laboratory values are: cholesterol in high-density lipoproteins (HDL-C), cholesterol (CH), predicted glomerular filtration rate (GFR), creatinine (C), urea (U), glucose (G), hemoglobin A_{1c} (HBA1C), prothrombin time (PT), hematocrit (HCT), mean corpuscular hemoglobin (MCH), c-reactive protein (CRP), oxygen (O), chloride (CHL).

importance. Therefore, only the three most important features for each classifier are listed in Table 4.3. The table with the diagnoses is provided in Section 3.1.3. For a more comprehensive overview of the feature importance, refer to Appendix B, where figures with the ten most important features are provided for each diagnosis. The first notable observation in the table is that age is present for all diagnoses except for one. This can be easily explained, as certain diagnoses are more likely to occur in specific age ranges or are, per definition, restricted to certain ages. For example, P39 refers to infections specific to the perinatal period, meaning that only infants younger than seven days are included with this diagnosis.

The case of P39 also illustrates an issue of using the same features for all the diagnoses. For certain diagnoses, the inclusion of age can be justified because there are different guidelines for different age groups as to what constitutes a healthy value for a particular feature. One such example is cholesterol [29]. By including age as a feature, the model might be able to differentiate the two classes more accurately. For other diagnoses, it might be more sensible to exclude samples from a certain age range and not use age as a feature. An analogous argument against the inclusion of gender can be made for pregnancy (O09).

In Table 4.1, one can see that the model for pregnancy performs best in all metrics except for accuracy. The inclusion of gender is insufficient to explain why that is. If that were the sole reason, one would expect a similarly high performance with P39. But in Table 4.3, age is the feature with the highest gain for the pregnancy classifier. Hence, it is possible that the combination of age and gender is sufficient

Diagnosis	Training Dataset				Test Dataset			
	Acc	Prec	Rec	F_1	Acc	Prec	Rec	F_1
I50	0.952	0.557	0.702	0.621	0.924	0.345	0.384	0.364

Table 4.4: **Comparison of performance on training and test dataset after the new split.** Accuracy (Acc), Precision (Prec), Recall (Rec), and F_1-score (F_1) are shown.

for the model to perform significantly better than the models for the other diagnoses.

Up to this point, the importance of other features has been neglected. The majority of the features in the dataset are laboratory blood values. Cholesterol in high-density lipoproteins (HDL-C) is listed in Table 4.3 for chronic ischemic heart disease (I25), essential hypertension (I10), and heart failure (I50). HDL-C is an effective predictor for cardiovascular disease [30]. For essential hypertension, serum cholesterol (CH) is also listed. Serum cholesterol has been identified as a predictor of essential hypertension [31]. Pregnancy (O09) impacts coagulation and hemodynamics, which includes prothrombin time [32]. For other diagnoses the features listed in Table 4.3 are part of their respective diagnostic criteria. This includes blood glucose and hemoglobin A_{1c} for diabetes (E11) [33], estimated glomerular filtration rate for chronic kidney disease (N18) [34], and both hematocrit and mean corpuscular hemoglobin for iron deficiency anemia (D50) [35]. This leads to the conclusion that there is a medical rationale behind the feature importance for most diagnoses, further refuting the possibility mentioned in Section 4.1.1 that there is little to no correlation between diagnosis and features.

4.2 Further Aspects

This section provides further investigation of certain aspects from the results described in Sections 4.1.1 and 4.1.2.

4.2.1 Further Analysis for Heart Failure

As seen in Section 4.1.2, the classifier for heart failure overperforms on the training dataset. The hyperparameter search and training steps are rerun based on a new split to find out if this is due to an unfortunate split between the training and test data sets. Furthermore, the hyperparameter ranges are investigated to rule out the possibility that the optimal parameters lie outside the chosen range.

Comparing the entry for heart failure (I50) in Table 4.2 and Table 4.4 the differ-

Hyperparameter	Original Split	New Split
learning_rate	0.174	0.020
scale_pos_weight	14.000	5.0
gamma	3.885	0.670
max_depth	14.000	11.000
min_child_weight	7.000	2.000
colsample_bytree	0.937	0.793
reg_alpha	1.804	0.186
reg_lambda	0.289	3.712

Table 4.5: **Comparison of hyperparameters between splits.**

ence is evident. In Table 4.4 with the new split, the performance on the training dataset is worse than that of the original split. However, performance on the test dataset improves over the first split. This means that although performance on the test dataset itself is worse comparatively, due to the model being less overfitted, performance on the actual test dataset is improved. Hence, the degree to which the classifier for heart failure overperformed on the first split is likely coincidental and caused by the specific split. Another possibility is that the choice of hyperparameters differs between the splits.

Table 4.5 shows a comparison of the selected hyperparameters between the two splits. For an explanation of the hyperparameters themselves, refer to Section 3.2. Notable is that almost all of the hyperparameters are dissimilar in value when comparing the splits. In part, this can be attributed to using a Bayesian method for the hyperparameter search. Since no specific values are given to choose from, there will be differences simply due to the generation of different random numbers. Furthermore, certain hyperparameters function similarly and thus have complementary effects. For example, both reg_alpha and reg_lambda are regularization parameters that scale the same variable. Parameters making the model less conservative, i.e., more prone to overfitting, might cause the more significant overfitting with the original split. The most likely is learning_rate, which is more than eight times higher than in the newer split. A higher learning rate increases the impact of newly added trees on the overall decision so that minor errors caused by variation in the training data are overcorrected.

The large difference between the performance on the training and test data is likely due to either or both of the above reasons and therefore a result of chance and not of some underlying structure in the data.

Diagnosis	Accuracy Classifier				Precision Classifier			
	Acc	Prec	Rec	F_1	Acc	Prec	Rec	F_1
I25	0.921	0.757	0.456	0.569	0.921	0.748	0.464	0.572
N18	0.932	0.620	0.167	0.263	0.925	0.483	0.456	0.469
O09	0.983	0.736	0.791	0.763	0.983	0.729	0.810	0.767

Table 4.6: **Performance on test dataset after using different target metrics for hyperparameter selection.** Accuracy (Acc), Precision (Prec), Recall (Rec), and F_1-score (F_1) are shown for both the classifier where accuracy was used as a target metric, and the classifier where precision was used as a target metric.

4.2.2 Different Evaluation Metrics for Hyperparameter Tuning

In Section 3.2 the procedure for obtaining optimal hyperparameters was discussed. The F_1-score is used as a metric to evaluate the different hyperparameter combinations. Using different evaluation metrics, the hyperparameters for which performance is best might change. Therefore, hyperparameter selection was performed twice with different target evaluation metrics, once with accuracy and once with precision. Instead of using all diagnoses, this procedure was limited to the diagnoses where the classifiers performed best with regards to the F_1-score.

Table 4.6 shows the performance with the two new target evaluation metrics being used. When using accuracy for the hyperparameter selection, the models are more accurate compared to when the F_1-score was used (see Table 4.2). The improvement is rather small, with the difference for pregnancy (O09) only being 0.001. There also is an improvement in precision, but simultaneously a decrease in recall leading to an overall decrease in the F_1-score.

When using precision in the hyperparameter search performance improves in that regard over the performance when the F_1-score was used. Accuracy also increases, but recall drops off, simultaneously decreasing the F_1-score. Quite surprising is that although there is an improvement in precision over the original performance, using accuracy as the metric leads to a more significant increase. One explanation is that the hyperparameter search is not exhaustive enough. Not using enough iterations in the search might cause suboptimal parameters to be chosen. However, this explanation seems less likely since the accuracy classifier performs better regarding precision for all three diagnoses. Hence, a more plausible cause is that a high precision measure in the hyperparameter search does not translate as well into high precision on the test dataset.

In conclusion, using the F_1-score for tuning improves the model's capability to distinguish between the two classes compared to when accuracy or precision are used.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

In this thesis, binary classification is applied for the first time to a dataset created for generating personalized reference intervals. The structure of the dataset places limitations on the feasibility of any classification task aiming at predicting diagnoses. This is partially circumvented by approximating cases based on available patient information.

Performance is assessed primarily with the F_1-score. Of the ten diagnoses studied, classification for pregnancy yielded the highest F_1-score. However, overall performance was poor compared to similar classification tasks in the literature. Class imbalance is one of the main reasons for the issues regarding performance. Another possible reason that cannot be dismissed is that only previously established diagnoses were collected. Because of this, the correlation between diagnoses and laboratory test measurements may be reduced, which would have an impact not only on classification but also on other applications of the dataset.

Nevertheless, the assessment of which features are relevant to specific diagnoses establishes a base correlation between some of the examined diagnoses and relevant features. This is an important confirmation of the suitability of the dataset for machine learning applications. In the subsequent section, possible measures to improve performance in classification tasks are discussed. Moreover, the section will briefly explore other possible avenues of future research.

5.2 Future Work

5.2.1 Addressing Limitations

As mentioned above, class imbalance is a significant hindrance to the model distinguishing the two classes. This is not only a problem with this specific thesis but with medical data in general since they are typically imbalanced. There are several ways of dealing with class imbalance, among which the most studied approaches address the issue at an algorithmic or data level [36]. One example of using algorithmic methods is the scale_pos_weight hyperparameter provided by XGBoost. No data-based approach has been applied in this thesis. One way of doing so is to either over- or undersample the training set. Oversampling involves generating new data points for the minority class, whereas undersampling reduces the number of data in the majority class. If such methods are to be used, undersampling should be preferred [37].

From the comparison of performance between training and test data, some degree of overfitting was evident. The work fails to provide a reliable estimate of the generalization error caused by overfitting, with the only estimate being a single comparison. A common way of obtaining a better estimate is to perform cross-validation on the training set to obtain multiple values and perform statistical analysis. In addition, to address the problem of overfitting more generally, one can choose not to optimize certain hyperparameters and instead opt for fixed values. Such hyperparameters could include the max_depth parameter or regularization parameters such as reg_alpha and reg_lambda.

One part of the results that might be caused by one of the limitations of the work also warrants further investigation. Classification for pregnancy yielded the best performance in all metrics but accuracy. As discussed in Section 4.1.3, the inclusion of gender for pregnancy is redundant. To ascertain whether the difference in performance is simply due to a combination of gender and age and not an underlying aspect of the data, such as the immediacy of the blood test to the diagnosis, classification should be repeated with different conditions. For example, retraining and testing the model on data where entries from males have been excluded.

Another limitation not previously discussed is the approach of this work itself. Because of its exploratory nature, the focus is not on specific diagnoses. Restricting the research to a specific diagnosis would allow the emphasis to be placed more carefully, for example, by performing feature selection or incorporating domain-specific knowledge from specialists for a particular diagnosis. Such measures are certainly possible with approaches similar to the one used in this work, but the effort required increases with the number of diagnoses studied, reducing feasibility.

5.2.2 Extending the Scope

Beyond using binary classification, other classification methods might be suitable for the specific dataset used in this work. One such method is described by Hansen, Sagi, and Hose, who use a Graph Convolutional Network-based approach for multi-label diagnostic prediction of electronic health records [38]. They introduce a novel method of using pre-initialized graph node embeddings derived from hierarchical medical taxonomies, such as LOINC and the ICD-9 diagnostic codification. The use of semantic data contained in the taxonomies in that way is also applicable to the data used in this work, making it an appropriate method for multiple classification. Moreover, graph-based data representations could potentially eliminate the need for grouping the data by patient as described in this thesis.

Helpful with any classification task would be a more extensive exploratory analysis. For example, performing clustering on positive cases for a specific diagnosis could help identify distinct data clusters. Excluding some of these clusters for training could improve performance since these clusters might correspond to patients who have received treatment for the illness. Medical experts would have to be consulted to identify clusters with relevant laboratory measurements.

Lastly, linking the data to other datasets might provide helpful insight into the progression and treatment of diseases. This is, however, difficult to implement since it would require renewed patient consent and approval from an ethics committee, among other complications. A more straightforward approach is extending research to a larger fraction of the entire Swiss BioRef data. This would still require approval from the respective data provider institutes and impose some technical hurdles, but it is still more feasible than linking other forms of data. Especially if a successful machine learning application is developed on the data from the University Hospital Bern, other data providers might be more inclined to approve the use of their data.

Appendix A

Supplementary Tables

Feature	Description
SubjectPseudoIdentifier	Patient identifier
AdministrativeCase	Case identifier
DataProviderInstitute	Data provider identifier
Labtest	Description of the laboratory test
LOINC	LOINC code corresponding to the test
LabResultValue	Measured value
LabResultUnit	Unit of measurement for the corresponding value
Diag01	ICD-10-GM code of relevant diagnosis 1
Diag02	ICD-10-GM code of relevant diagnosis 2
Diag03	ICD-10-GM code of relevant diagnosis 3
Diag04	ICD-10-GM code of relevant diagnosis 4
Diag05	ICD-10-GM code of relevant diagnosis 5
Age	Patient's age
AgeUnit	Age unit
AgeType	LOINC identifying the age type
AdministrativeGender	Patient's administrative gender
device_udi	Unique device identifier from the Global Unique Device Identification Database
device_type	Device type identifiers from the Global Medical Device Nomenclature
testkit_udi	Unique testkit identifier from the Global Unique Device Identification Database
testkit_type	Testkit type identifiers from the Global Medical Device Nomenclature
RFIKey_device	-

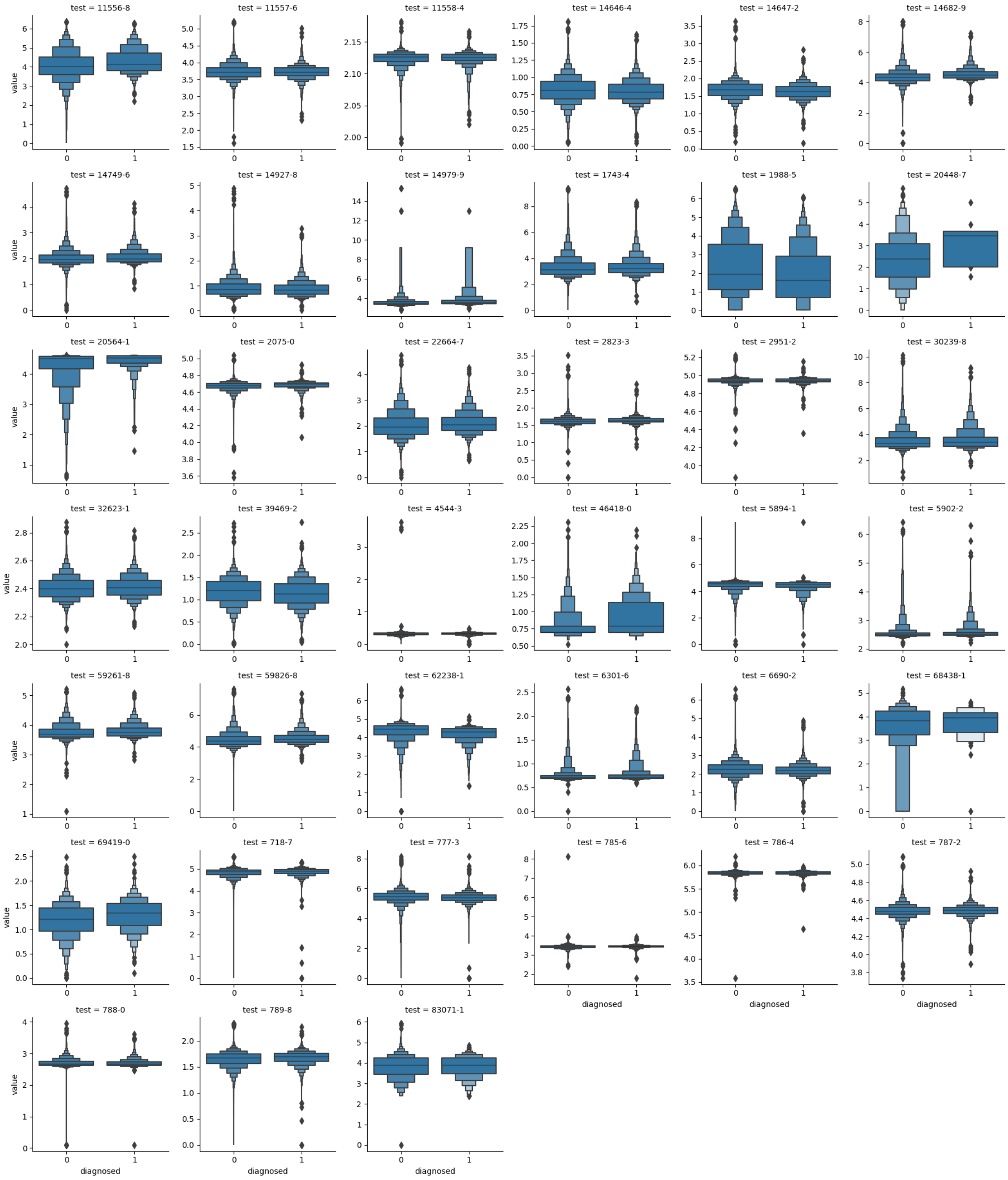
Table A.1: **Complete list of features present in the dataset**

LOINC	Description
2823-3	Potassium [Moles/volume] in Serum or Plasma
2951-2	Sodium [Moles/volume] in Serum or Plasma
5894-1	Prothrombin time (PT) actual/Normal
1988-5	C reactive protein [Mass/volume] in Serum or Plasma
14749-6	Glucose [Moles/volume] in Serum or Plasma
14682-9	Creatinine [Moles/volume] in Serum or Plasma
6690-2	Leukocytes [#/volume] in Blood by Automated count
777-3	Platelets [#/volume] in Blood by Automated count
718-7	Hemoglobin [Mass/volume] in Blood
789-8	Erythrocytes [#/volume] in Blood by Automated count
4544-3	Hematocrit [Volume Fraction] of Blood by Automated count
787-2	MCV [Entitic volume] by Automated count
785-6	MCH [Entitic mass] by Automated count
786-4	MCHC [Mass/volume] by Automated count
6301-6	INR in Platelet poor plasma by Coagulation assay
788-0	Erythrocyte distribution width [Ratio] by Automated count
32623-1	Platelet mean volume [Entitic volume] in Blood by Automated count
62238-1	Glomerular filtration rate/1.73 sq M.predicted [Volume Rate/Area] in Serum, Plasma or Blood by Creatinine-based formula (CKD-EPI)
5902-2	Prothrombin time (PT)
1743-4	Alanine aminotransferase [Enzymatic activity/volume] in Serum or Plasma by With P-5'-P
30239-8	Aspartate aminotransferase [Enzymatic activity/volume] in Serum or Plasma by With P-5'-P
22664-7	Urea [Moles/volume] in Serum or Plasma
11558-4	pH of Blood
11557-6	Carbon dioxide [Partial pressure] in Blood
11556-8	Oxygen [Partial pressure] in Blood
2075-0	Chloride [Moles/volume] in Serum or Plasma
14979-9	aPTT in Platelet poor plasma by Coagulation assay
20564-1	Oxygen saturation in Blood
59826-8	Creatinine [Moles/volume] in Blood
14927-8	Triglyceride [Moles/volume] in Serum or Plasma
14647-2	Cholesterol [Moles/volume] in Serum or Plasma
14646-4	Cholesterol in HDL [Moles/volume] in Serum or Plasma
39469-2	Cholesterol in LDL [Moles/volume] in Serum or Plasma by calculation
4548-4	Hemoglobin A1c/Hemoglobin.total in Blood by IFCC protocol
46418-0	INR in Capillary blood by Coagulation assay
83071-1	25-Hydroxyvitamin D2+25-Hydroxyvitamin D3 [Moles/volume] in Serum or Plasma by Immunoassay
69419-0	Cholesterol in LDL [Moles/volume] in Serum or Plasma by Direct assay
62292-8	25-Hydroxyvitamin D2+25-Hydroxyvitamin D3 [Moles/volume] in Serum or Plasma
20448-7	Insulin [Units/volume] in Serum or Plasma

Table A.2: **Complete list of laboratory tests present in the dataset**

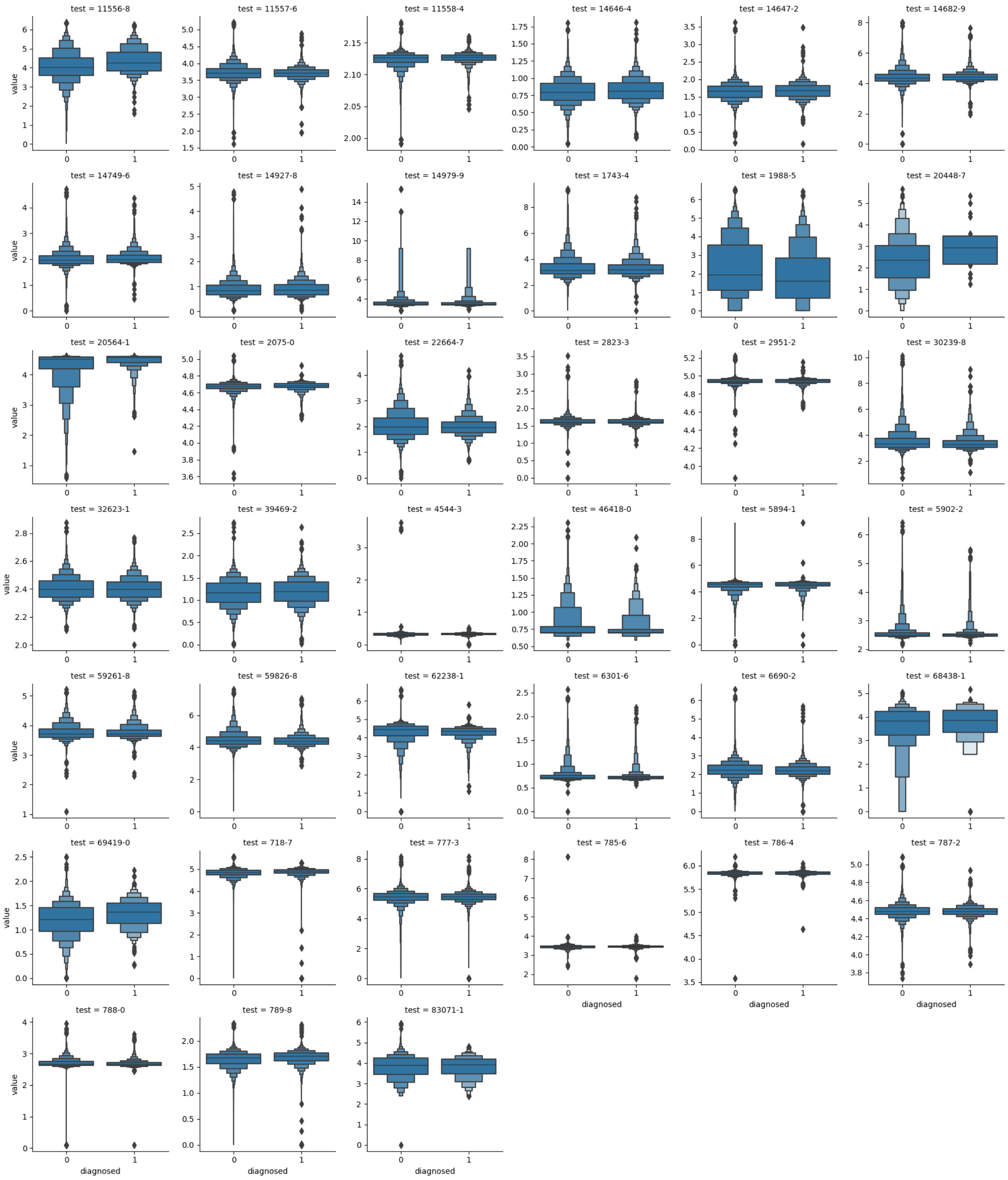
Appendix B

Supplementary Figures



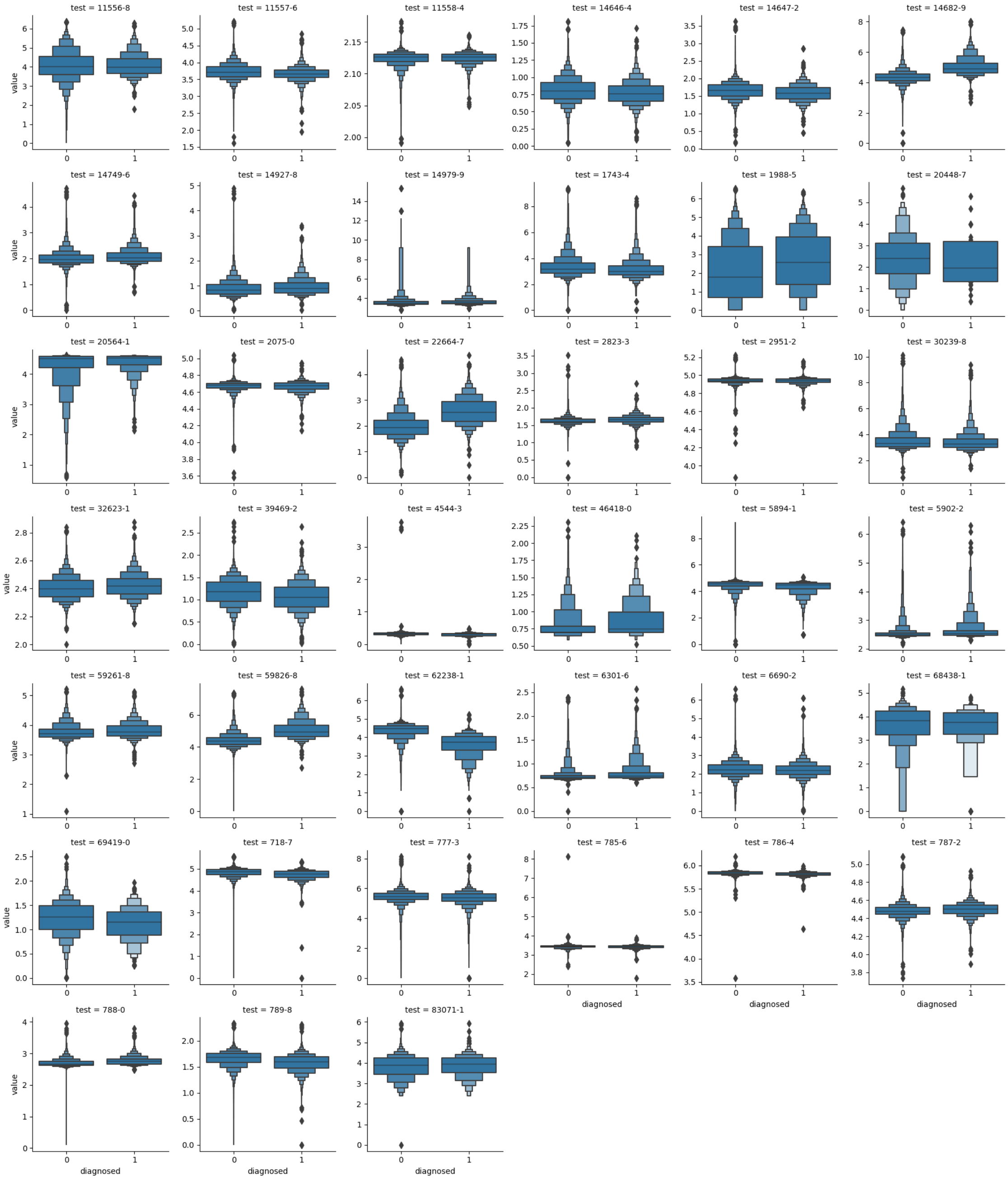
(a) I25

Figure B.1: **Comparison of measured values between patients with and without the diagnosis.** To account for outliers, values were shifted up by one, and the logarithm was taken. Logarithmic values are on the y-axis. The x-axis indicates the patient group; (1) diagnosis present and (0) diagnosis absent.



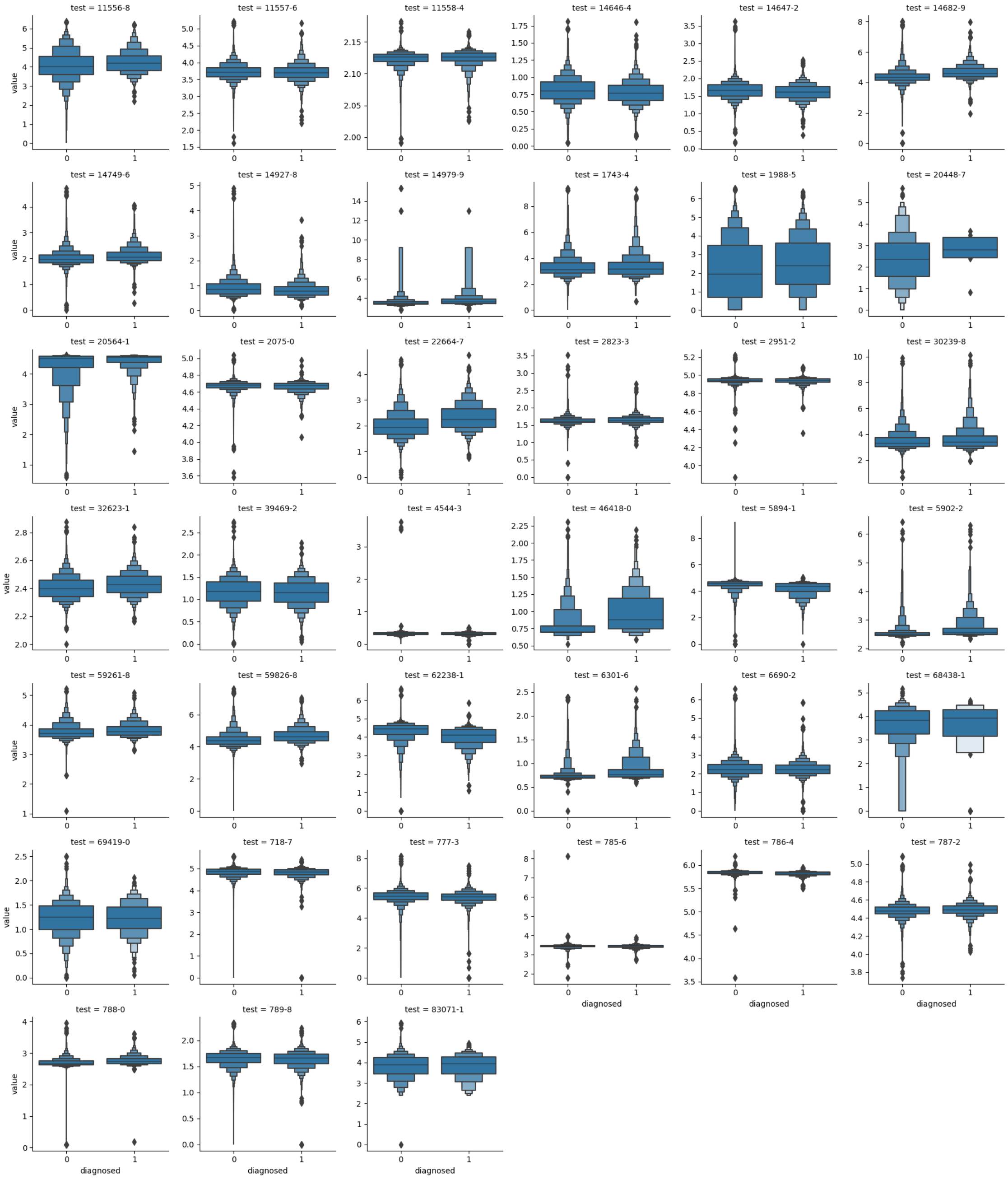
(b) I10

Figure B.1: **Comparison of measured values between patients with and without the diagnosis.** To account for outliers, values were shifted up by one, and the logarithm was taken. Logarithmic values are on the y-axis. The x-axis indicates the patient group; (1) diagnosis present and (0) diagnosis absent.



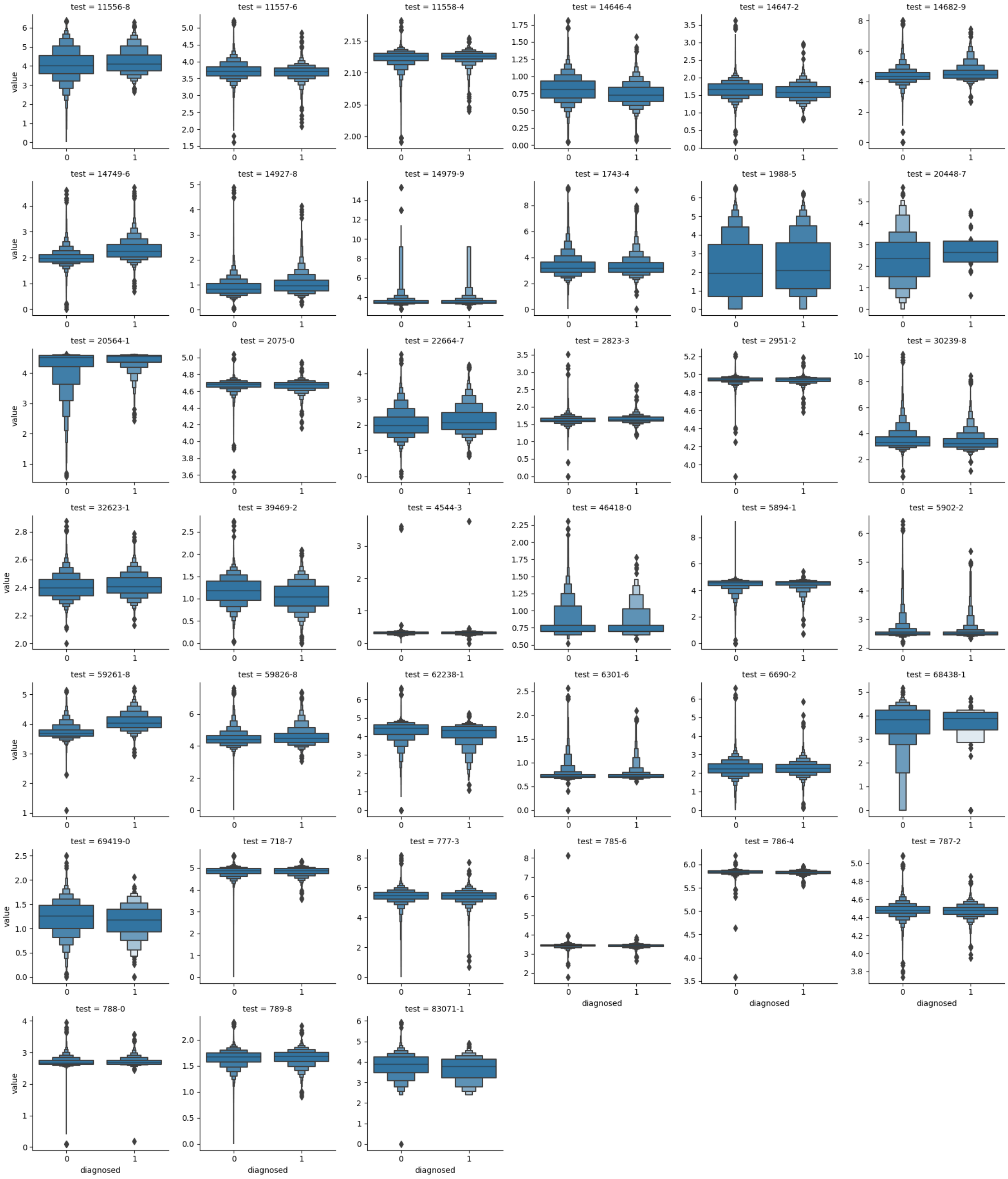
(c) N18

Figure B.1: **Comparison of measured values between patients with and without the diagnosis.** To account for outliers, values were shifted up by one, and the logarithm was taken. Logarithmic values are on the y-axis. The x-axis indicates the patient group; (1) diagnosis present and (0) diagnosis absent.



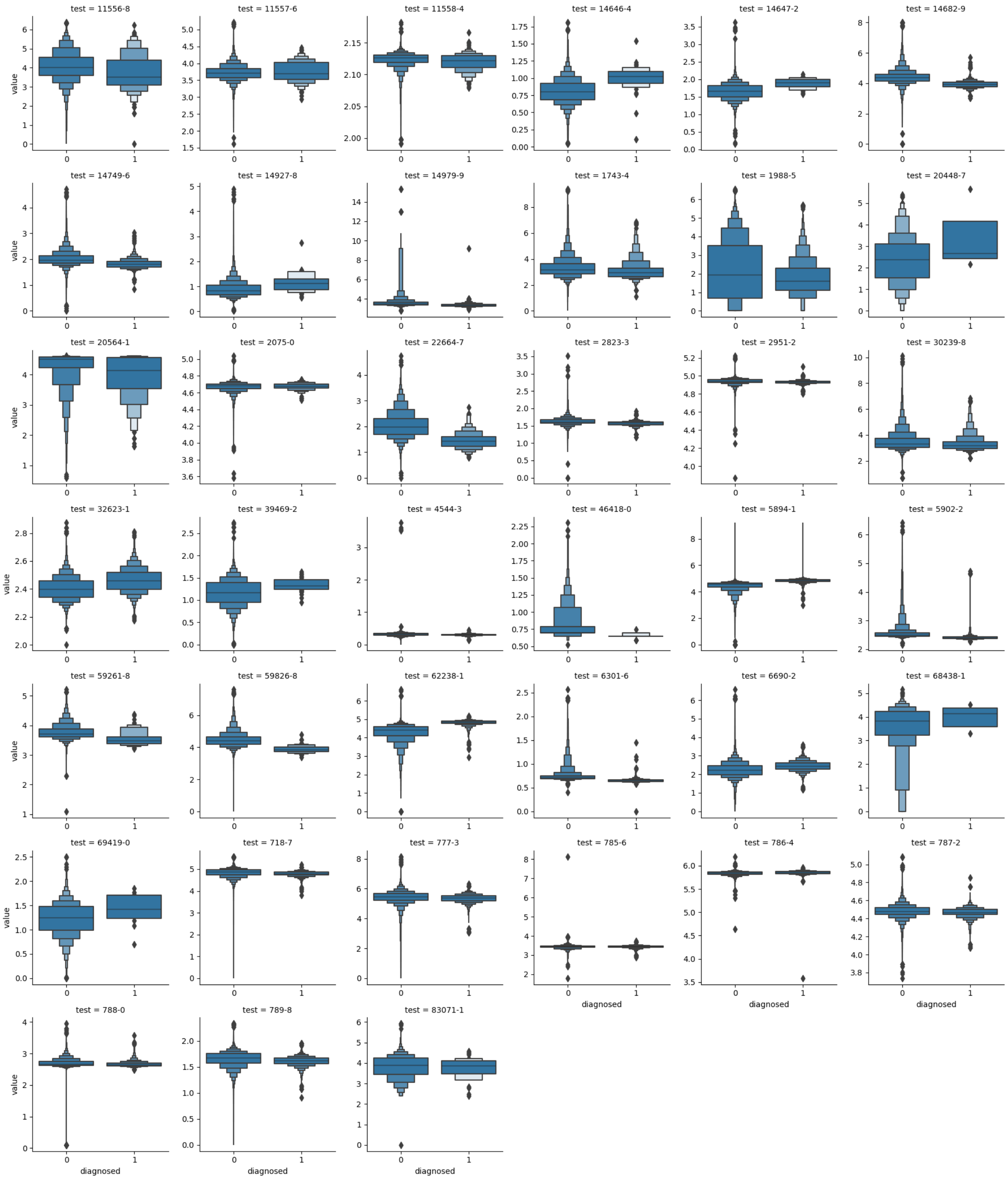
(d) I50

Figure B.1: **Comparison of measured values between patients with and without the diagnosis.** To account for outliers, values were shifted up by one, and the logarithm was taken. Logarithmic values are on the y-axis. The x-axis indicates the patient group; (1) diagnosis present and (0) diagnosis absent.



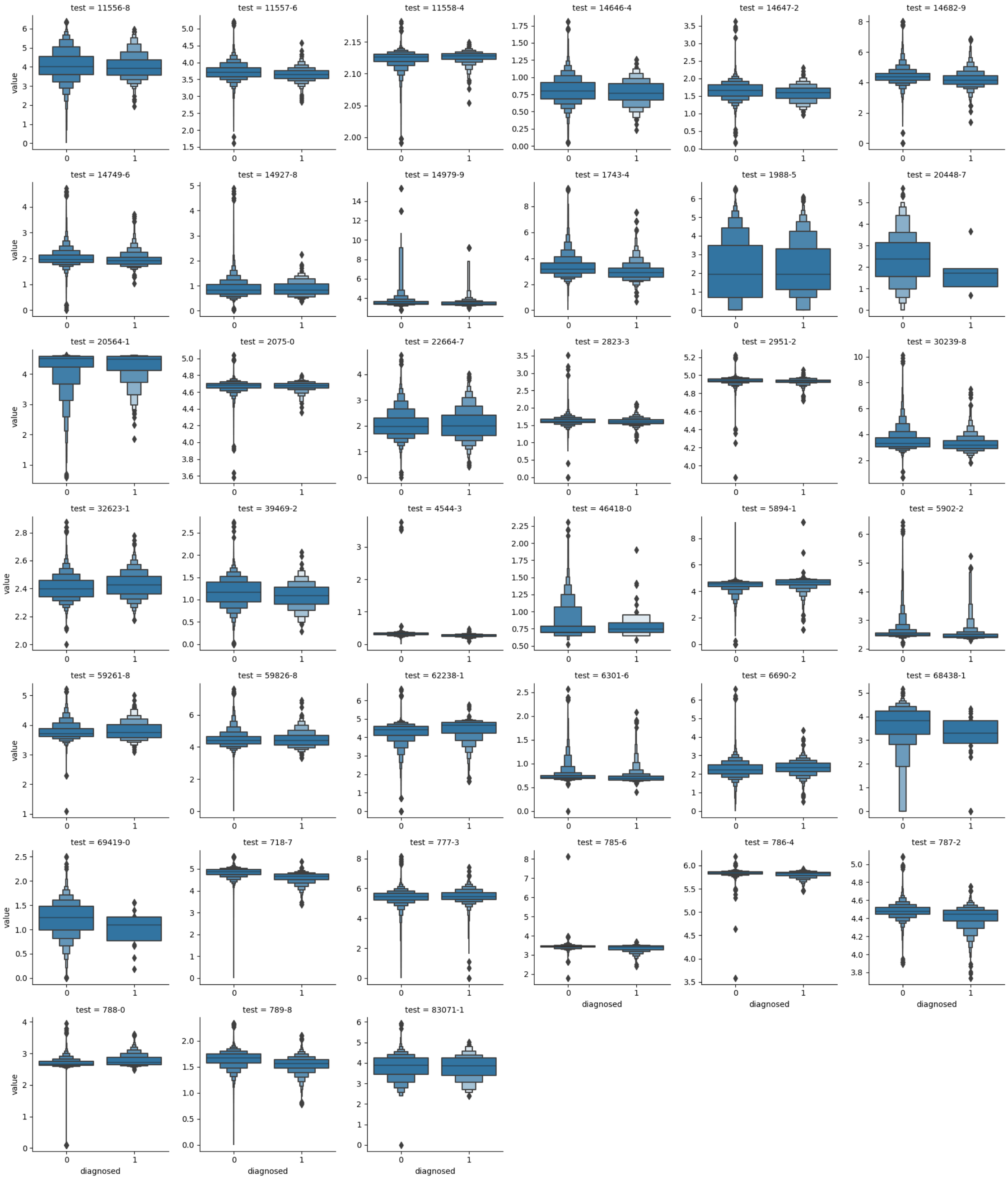
(e) E11

Figure B.1: **Comparison of measured values between patients with and without the diagnosis.** To account for outliers, values were shifted up by one, and the logarithm was taken. Logarithmic values are on the y-axis. The x-axis indicates the patient group; (1) diagnosis present and (0) diagnosis absent.



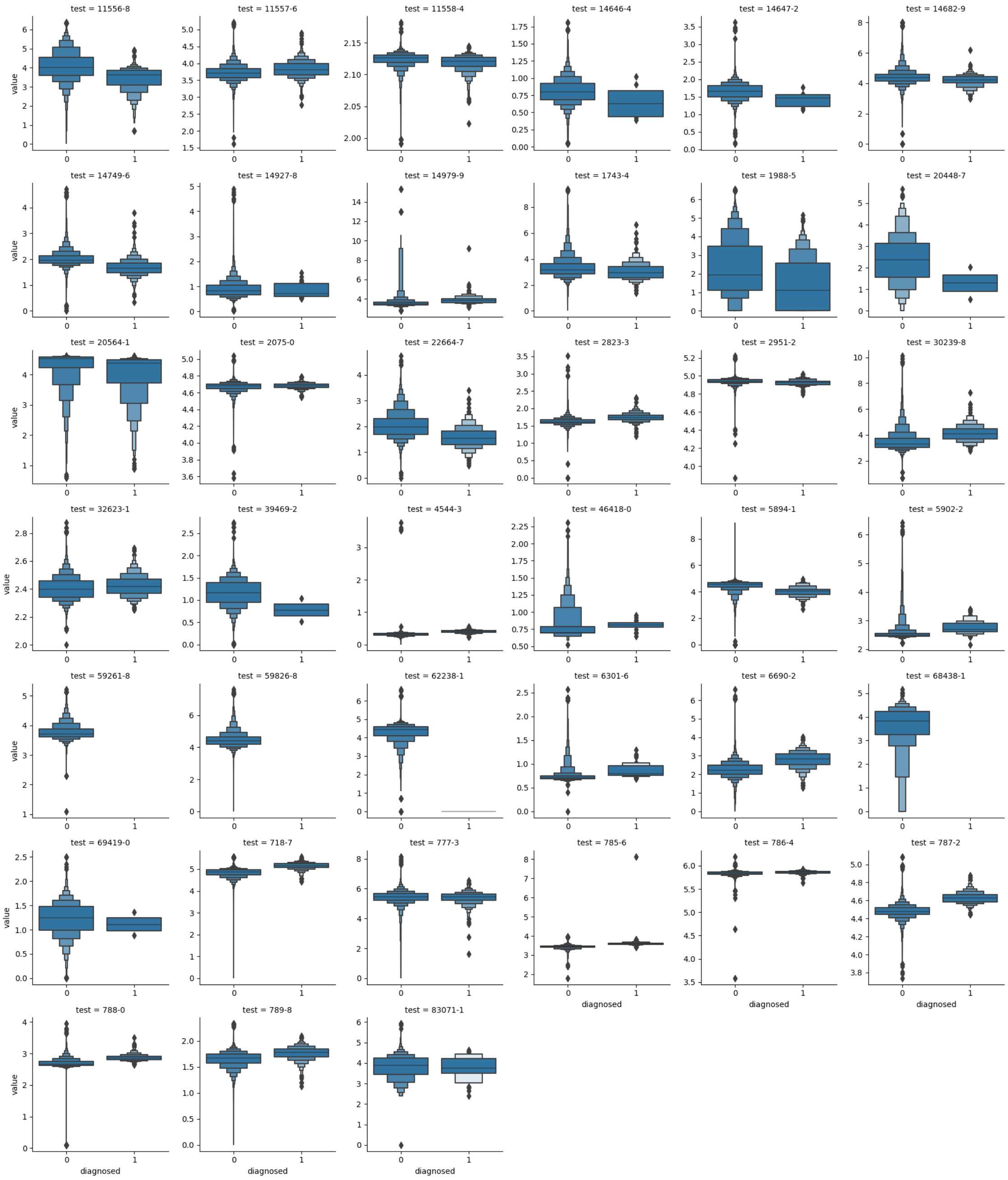
(f) O09

Figure B.1: **Comparison of measured values between patients with and without the diagnosis.** To account for outliers, values were shifted up by one, and the logarithm was taken. Logarithmic values are on the y-axis. The x-axis indicates the patient group; (1) diagnosis present and (0) diagnosis absent.



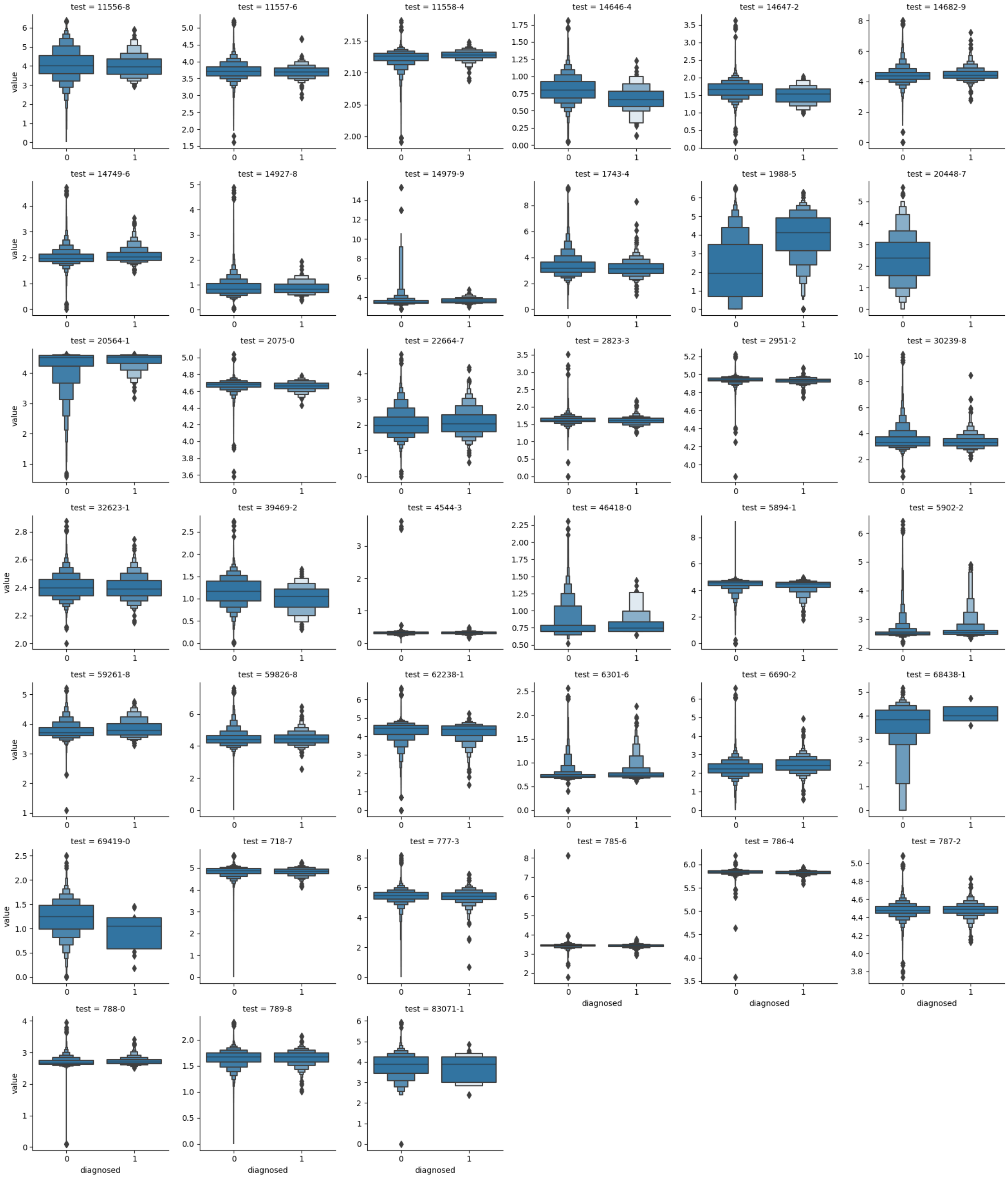
(g) D50

Figure B.1: **Comparison of measured values between patients with and without the diagnosis.** To account for outliers, values were shifted up by one, and the logarithm was taken. Logarithmic values are on the y-axis. The x-axis indicates the patient group; (1) diagnosis present and (0) diagnosis absent.



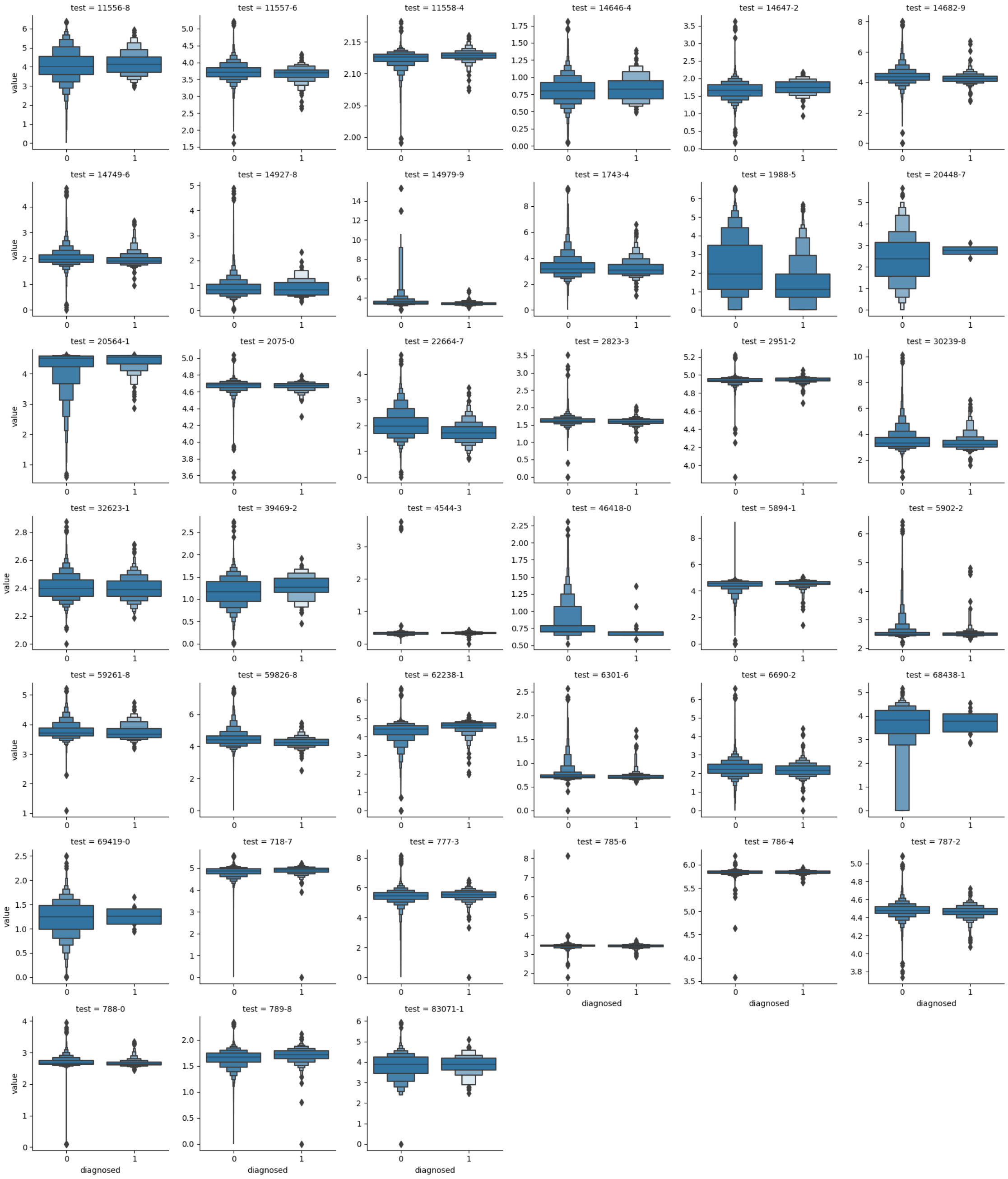
(h) P39

Figure B.1: **Comparison of measured values between patients with and without the diagnosis.** To account for outliers, values were shifted up by one and the logarithm was taken. Logarithmic values are on the y-axis. The x-axis indicates the patient group; (1) diagnosis present and (0) diagnosis absent.



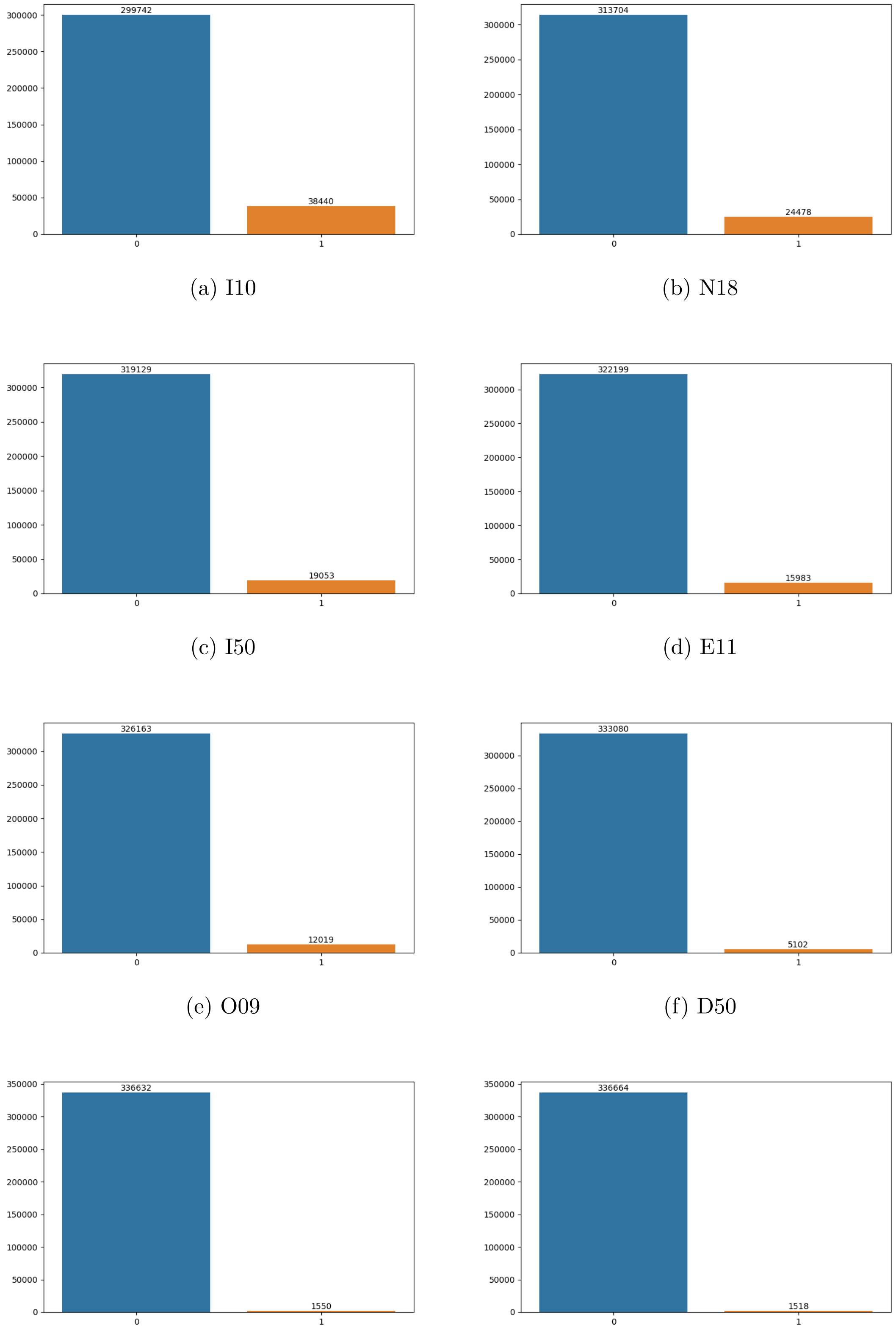
(i) A46

Figure B.1: **Comparison of measured values between patients with and without the diagnosis.** To account for outliers, values were shifted up by one and the logarithm was taken. Logarithmic values are on the y-axis. The x-axis indicates the patient group; (1) diagnosis present and (0) diagnosis absent.



(j) F43

Figure B.1: **Comparison of measured values between patients with and without the diagnosis.** To account for outliers, values were shifted up by one and the logarithm was taken. Logarithmic values are on the y-axis. The x-axis indicates the patient group; (1) diagnosis present and (0) diagnosis absent.



(a) I10

(b) N18

(c) I50

(d) E11

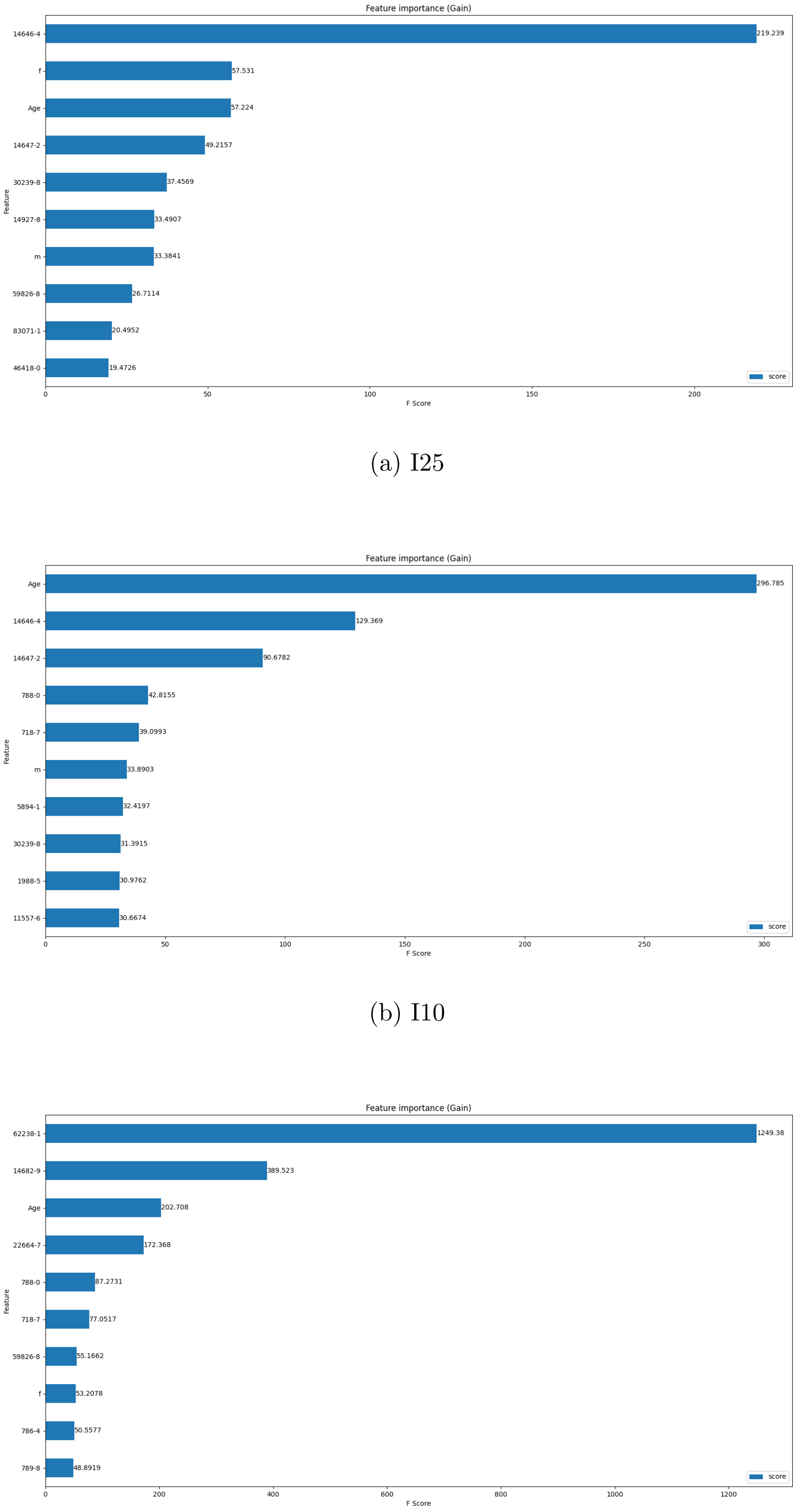
(e) O09

(f) D50

(g) P39

(h) A46

Figure B.2: **Class imbalance.** Number of cases with (1) and without (0) diagnosis.

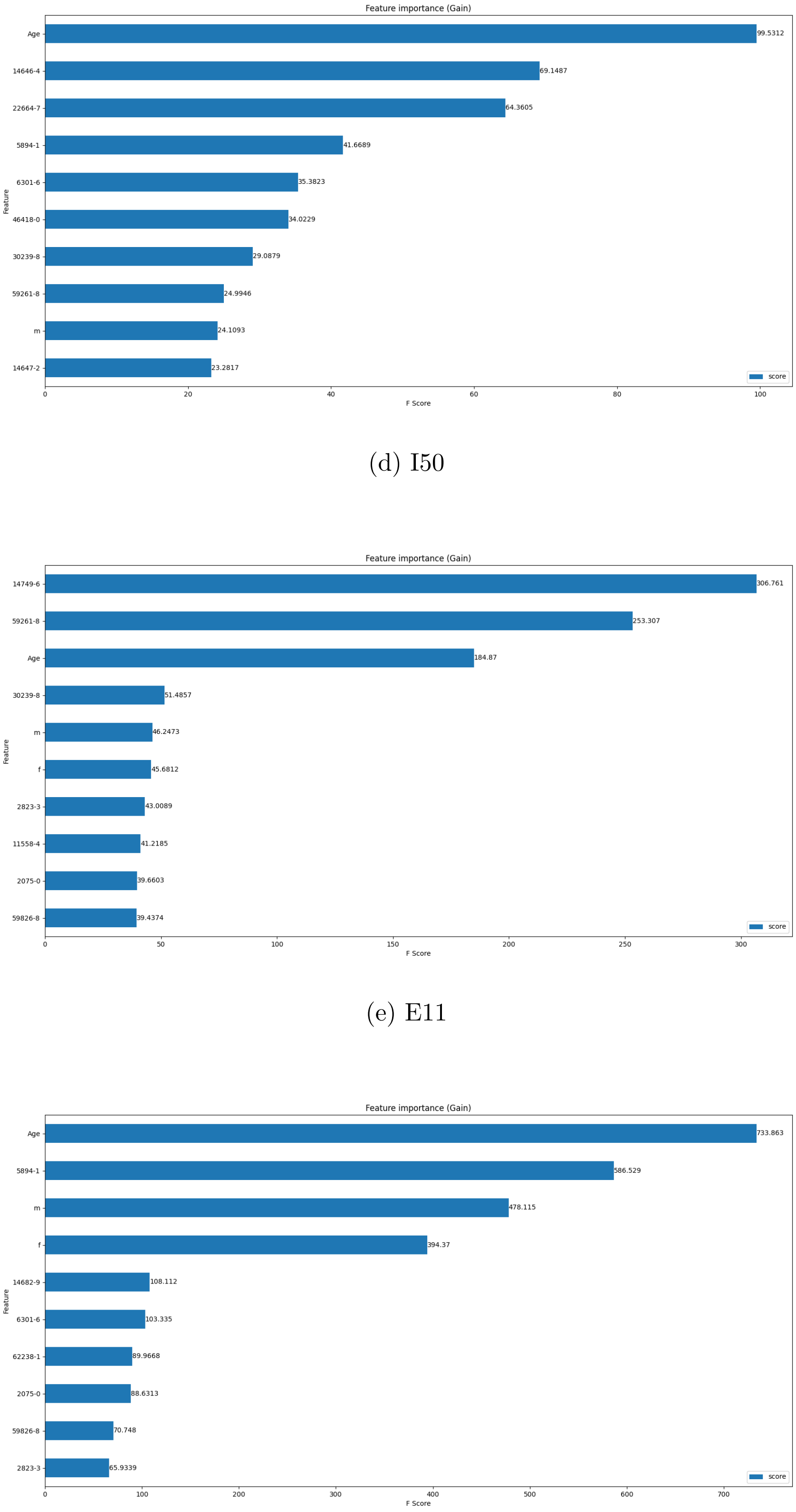


(a) I25

(b) I10

(c) N18

Figure B.3: **Ten most important features for each diagnosis according to gain.** Diagnoses are indicated by their ICD-10-GM codes.

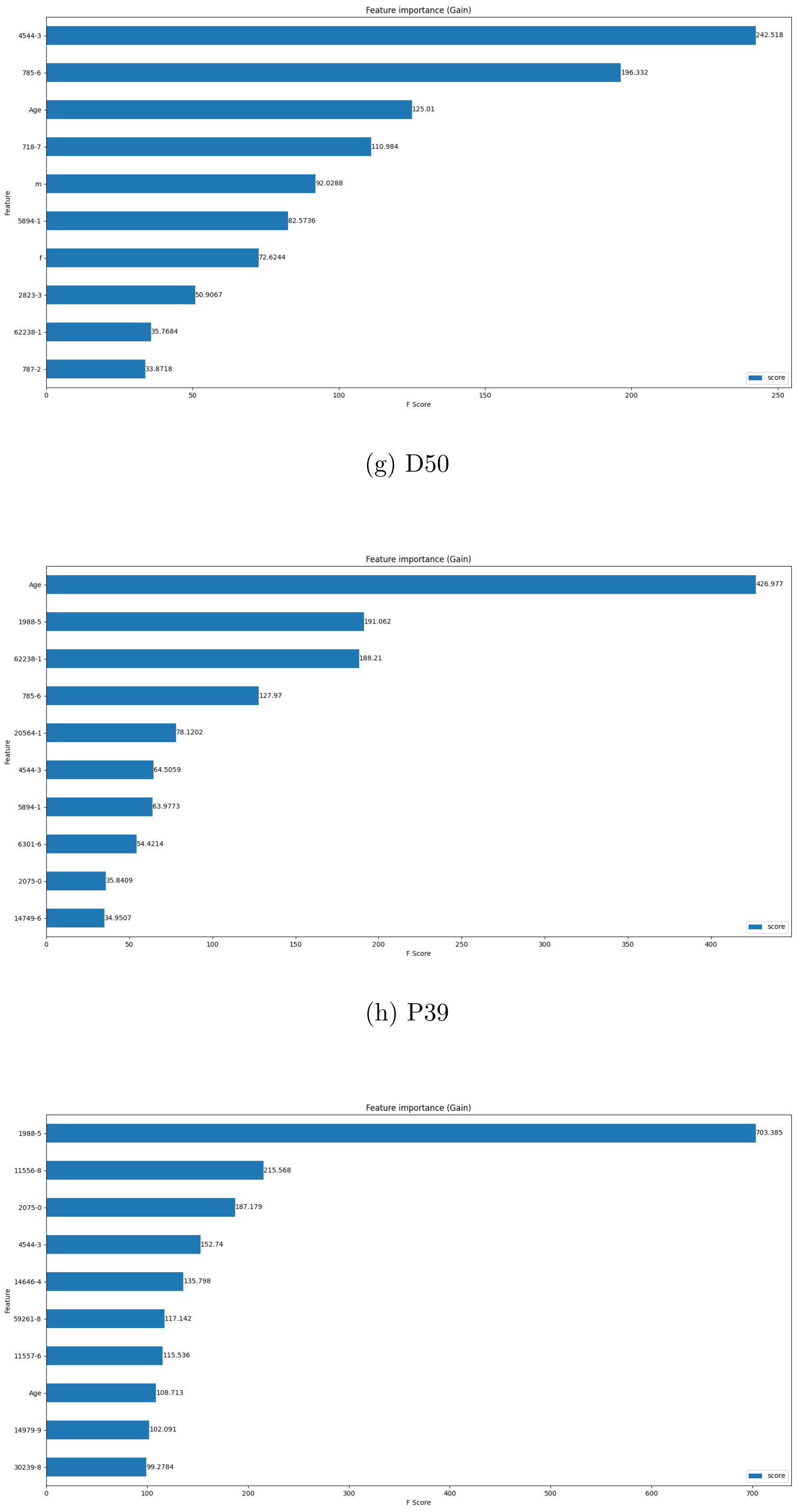


(d) I50

(e) E11

(f) O09

Figure B.3: **Ten most important features for each diagnosis according to gain.** Diagnoses are indicated by their ICD-10-GM codes.

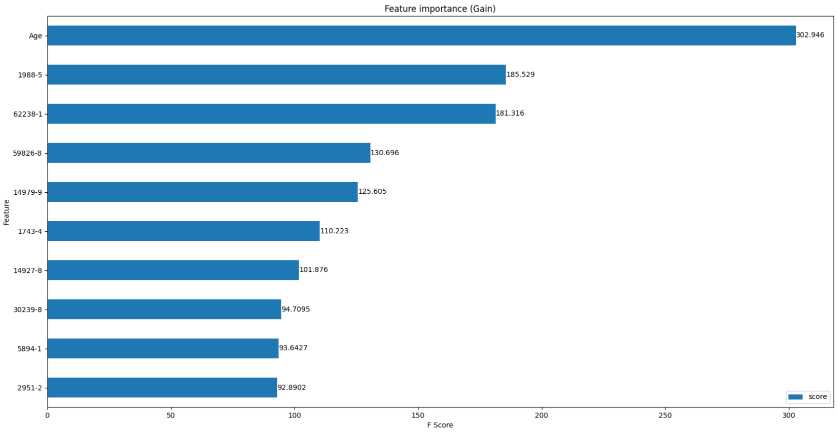


(g) D50

(h) P39

(i) A46

Figure B.3: **Ten most important features for each diagnosis according to gain.** Diagnoses are indicated by their ICD-10-GM codes.



(j) F43

Figure B.3: **Ten most important features for each diagnosis according to gain.** Diagnoses are indicated by their ICD-10-GM codes.

Bibliography

[1] Igor Kononenko. Machine learning for medical diagnosis: history, state of the art and perspective. *Artif. Intell. Medicine*, 23(1):89–109, 2001.

[2] Mohammad Shehab, Laith Abualigah, Qusai Shambour, Muhannad A. Abu-Hashem, Moh'd Khaled Yousef Shambour, Ahmed Izzat Alsalibi, and Amir H. Gandomi. Machine learning in medical applications: A review of state-of-the-art methods. *Comput. Biol. Medicine*, 145:105458, 2022.

[3] A. Rajkomar, J. Dean, and I. Kohane. Machine Learning in Medicine. *N Engl J Med*, 380(14):1347–1358, Apr 2019.

[4] J. Vamathevan, D. Clark, P. Czodrowski, I. Dunham, E. Ferran, G. Lee, B. Li, A. Madabhushi, P. Shah, M. Spitzer, and S. Zhao. Applications of machine learning in drug discovery and development. *Nat Rev Drug Discov*, 18(6):463–477, Jun 2019.

[5] Kaustubh Arun Bhavsar, Jimmy Singla, Yasser D Al-Otaibi, Oh-Young Song, Yousaf Bin Zikria, and Ali Kashif Bashir. Medical diagnosis using machine learning: a statistical review. *Computers, Materials and Continua*, 67(1):107–125, 2021.

[6] M. E. Maros, D. Capper, D. T. W. Jones, V. Hovestadt, A. von Deimling, S. M. Pfister, A. Benner, M. Zucknick, and M. Sill. Machine learning workflows to estimate class probabilities for precision cancer diagnostics on DNA methylation microarray data. *Nat Protoc*, 15(2):479–512, Feb 2020.

[7] M. D. Abràmoff, P. T. Lavin, M. Birch, N. Shah, and J. C. Folk. Pivotal trial of an autonomous AI-based diagnostic system for detection of diabetic retinopathy in primary care offices. *NPJ Digit Med*, 1:39, 2018.

[8] Adeola Ogunleye and Qing-Guo Wang. Xgboost model for chronic kidney disease diagnosis. *IEEE ACM Trans. Comput. Biol. Bioinform.*, 17(6):2131–2140, 2020.

[9] Leila Ismail, Huned Materwala, Maryam Tayefi, Phuong Ngo, and Achim P. Karduck. Type 2 diabetes with artificial intelligence machine learning: Methods and evaluation. *Archives of Computational Methods in Engineering*, 29(1):313–333, Jan 2022.

[10] Iman Azarkhish, Mohammad Reza Raoufy, and Shahriar Gharibzadeh. Artificial intelligence models for predicting iron deficiency anemia and iron serum level based on accessible laboratory data. *J. Medical Syst.*, 36(3):2057–2061, 2012.

[11] Margarita Sakhibgareeva and A.Yu Zaozersky. Developing an artificial intelligence-based system for medical prediction. *Bulletin of Russian State Medical University*, 6:40–44, 11 2017.

[12] Tobias Ueli Blatter, Harald Witte, Jules Fasquelle-Lopez, Jean Louis Raisaro, and Alexander Benedikt Leichtle. Cohort Profile: Swiss BioRef: The building blocks of a nationwide IT infrastructure in Switzerland for generating precise reference intervals. *medRxiv*, pages 2022–08, 2022.

[13] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. *CoRR*, abs/1603.02754, 2016.

[14] G. J. Escobar, B. J. Turk, A. Ragins, J. Ha, B. Hoberman, S. M. LeVine, M. A. Ballesca, V. Liu, and P. Kipnis. Piloting electronic medical record-based early detection of inpatient deterioration in community hospitals. *J Hosp Med*, 11 Suppl 1(Suppl 1):S18–S24, Nov 2016.

[15] John R Ball and Erin Balogh. Improving diagnosis in health care: Highlights of a report from the national academies of sciences, engineering, and medicine. *Ann Intern Med*, 164(1):59–61, September 2015.

[16] J. W. H. Kocks, H. Cao, B. Holzhauer, A. Kaplan, J. M. FitzGerald, K. Kostikas, D. Price, H. K. Reddel, I. Tsiligianni, C. F. Vogelmeier, S. Bostel, and P. Mastoridis. Diagnostic Performance of a Machine Learning Algorithm (Asthma/Chronic Obstructive Pulmonary Disease [COPD] Differentiation Classification) Tool Versus Primary Care Physicians and Pulmonologists in Asthma, COPD, and Asthma/COPD Overlap. *J Allergy Clin Immunol Pract*, 11(5):1463–1474, May 2023.

[17] E. Vayena, A. Blasimme, and I. G. Cohen. Machine learning in medicine: Addressing ethical challenges. *PLoS Med*, 15(11):e1002689, Nov 2018.

[18] Zachary C. Lipton. The mythos of model interpretability. *ACM Queue*, 16(3):30, 2018.

[19] Leo Breiman, J. H. Friedman, Richard A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth, 1984.

[20] Jerome H Friedman. 999 reitz lecture greedy function approximation: A gradient boosting machine 1, 2001.

[21] Jules Fasquelle-Lopez and Jean Louis Raisaro. An ontology and data converter from RDF to the i2b2 data model. In Brigitte Séroussi, Patrick Weber, Ferdinand Dhombres, Cyril Grouin, Jan-David Liebe, Sylvia Pelayo, Andrea Pinna, Bastien Rance, Lucia Sacchi, Adrien Ugon, Arriel Benis, and Parisis Gallos, editors, *Challenges of Trustable AI and Added-Value on Health - Proceedings of MIE 2022, Medical Informatics Europe, Nice, France, May 27-30, 2022*, volume 294 of *Studies in Health Technology and Informatics*, pages 372–376. IOS Press, 2022.

[22] C. J. McDonald, S. M. Huff, J. G. Suico, G. Hill, D. Leavelle, R. Aller, A. Forrey, K. Mercer, G. DeMoor, J. Hook, W. Williams, J. Case, and P. Maloney. LOINC, a universal standard for identifying laboratory observations: a 5-year update. *Clin Chem*, 49(4):624–633, Apr 2003.

[23] B Graubner and T Auhuber. ICD-10-GM 2009, Systematisches Verzeichnis. *Internationale statistische Klassifikation der Krankheiten und verwandter Gesundheitsprobleme*, 10, 2005.

[24] Haibo He and Edwardo A. Garcia. Learning from imbalanced data. *IEEE Trans. Knowl. Data Eng.*, 21(9):1263–1284, 2009.

[25] James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *J. Mach. Learn. Res.*, 13:281–305, 2012.

[26] Kartik Budholiya, Shailendra Kumar Shrivastava, and Vivek Sharma. An optimized xgboost based diagnostic system for effective prediction of heart disease. *J. King Saud Univ. Comput. Inf. Sci.*, 34(7):4514–4523, 2022.

[27] James Bergstra, Daniel Yamins, and David D. Cox. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pages 115–123. JMLR.org, 2013.

[28] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake VanderPlas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.*, 12:2825–2830, 2011.

[29] R. B. Goldberg, N. J. Stone, and S. M. Grundy. The 2018 AHA/ACC/AACVPR/AAPA/ABC/ACPM/ADA/AGS/APhA/ASPC/NLA/PCNA Guidelines on the Management of Blood Cholesterol in Diabetes. *Diabetes Care*, 43(8):1673–1678, Aug 2020.

[30] D. J. Rader and G. K. Hovingh. HDL and cardiovascular disease. *Lancet*, 384(9943):618–625, Aug 2014.

[31] J. V. Selby, G. D. Friedman, and C. P. Quesenberry. Precursors of essential hypertension: pulmonary function, heart rate, uric acid, serum cholesterol, and other serum chemistries. *Am J Epidemiol*, 131(6):1017–1027, Jun 1990.

[32] C. Hui, M. Lili, C. Libin, Z. Rui, G. Fang, G. Ling, and Z. Jianping. Changes in coagulation and hemodynamics during pregnancy: a prospective longitudinal study of 58 cases. *Arch Gynecol Obstet*, 285(5):1231–1236, May 2012.

[33] American Diabetes Association. 2. classification and diagnosis of diabetes: standards of medical care in diabetes—2018. *Diabetes Care*, 41(Supplement_1):S13–S27, 2018.

[34] A. S. Levey, L. A. Stevens, C. H. Schmid, Y. L. Zhang, A. F. Castro, H. I. Feldman, J. W. Kusek, P. Eggers, F. Van Lente, T. Greene, and J. Coresh. A new equation to estimate glomerular filtration rate. *Ann Intern Med*, 150(9):604–612, May 2009.

[35] T. D. Johnson-Wimbley and D. Y. Graham. Diagnosis and management of iron deficiency anemia in the 21st century. *Therap Adv Gastroenterol*, 4(3):177–184, May 2011.

[36] Vicente García, José Salvador Sánchez, and Ramón Alberto Mollineda. On the effectiveness of preprocessing methods when dealing with different levels of class imbalance. *Knowl. Based Syst.*, 25(1):13–21, 2012.

[37] Ricardo Barandela, José Salvador Sánchez, Vicente García, and E. Rangel. Strategies for learning in class imbalance problems. *Pattern Recognit.*, 36(3):849–851, 2003.

[38] Emil Riis Hansen, Tomer Sagi, and Katja Hose. Diagnosis prediction over patient data using hierarchical medical taxonomies. In George Fletcher and Verena Kantere, editors, *Proceedings of the Workshops of the EDBT/ICDT 2023 Joint Conference, Ioannina, Greece, March, 28, 2023*, volume 3379 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2023.

Erklärung

gemäss Art. 30 RSL Phil.-nat.18

Name/Vorname: Gribi Fabian

Matrikelnummer: 19-940-410

Studiengang: Informatik

Bachelor ☑ Master □ Dissertation □

Titel der Arbeit: Binary Classification of Blood Values

LeiterIn der Arbeit: PD Dr. Kaspar Riesen

Ich erkläre hiermit, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen benutzt habe. Alle Stellen, die wörtlich oder sinngemäss aus Quellen entnommen wurden, habe ich als solche gekennzeichnet. Mir ist bekannt, dass andernfalls der Senat gemäss Artikel 36 Absatz 1 Buchstabe r des Gesetzes vom 5. September 1996 über die Universität zum Entzug des auf Grund dieser Arbeit verliehenen Titels berechtigt ist.
Für die Zwecke der Begutachtung und der Überprüfung der Einhaltung der Selbständigkeitserklärung bzw. der Reglemente betreffend Plagiate erteile ich der Universität Bern das Recht, die dazu erforderlichen Personendaten zu bearbeiten und Nutzungshandlungen vorzunehmen, insbesondere die schriftliche Arbeit zu vervielfältigen und dauerhaft in einer Datenbank zu speichern sowie diese zur Überprüfung von Arbeiten Dritter zu verwenden oder hierzu zur Verfügung zu stellen.

Bern, 21.08.2023

Ort/Datum

Unterschrift