University of Magdeburg

Faculty of Computer Science

Master Thesis

Visual Model Interpretation for Epidemiological Cohort Studies

Author:

Tim Sabsch

4th May 2018

Advisers:

Prof. Bernhard Preim

Department of Simulation and Graphics

Visualization Group

Prof. Myra Spiliopoulou

Uli Niemann

Department of Technical and Business Information Systems

Knowledge Management and Discovery Lab

Sabsch, Tim:
Visual Model Interpretation for Epidemiological Cohort Studies
Master Thesis, University of Magdeburg, 2018.

Abstract

Epidemiological cohort studies investigate the cause and development of diseases in human populations. Conventional analyses are challenged by recently increasing study sizes, which is why the incorporation of machine learning gains popularity. State-of-the-art classifiers are however often hard to interpret – an important requirement in medical applications. This thesis addresses the gap between predictive power and interpretability in the context of cohort study analysis. Main contribution is the development of an interactive visual interface for the interpretation and comparison of probabilistic classifiers. It supports the analysis of important features at both global and individual level, computation of partial dependence, and iterative construction of meaningful feature groups. To analyse the longitudinal influence of features, the user can modify the feature set by removing a feature or replacing its value by a previous examination record. The developed visual interface is evaluated in two case studies in order to test its effectiveness for the generation and validation of research hypotheses. The case studies include a real-world epidemiological cohort study and synthetic data. The results indicate the interface's usefulness for epidemiological research, but also reveal necessary further work for the deployment into a productive environment.

Acknowledgements

The presented thesis has been written at the Otto-von-Guericke-University Magdeburg under the supervision of Bernhard Preim, Myra Spiliopoulou and Uli Niemann.

I would like to thank Bernhard Preim for many meetings and discussions together throughout the project. As someone without a strong background in visualisation, I appreciate his expertise and guidance. His pace at giving feedback surprises me every time. I would also like to thank Myra Spiliopoulou for her feedback. Her emails, usually being sent at a late hour, were always very constructive and food for thoughts. Thank goes also to Uli Niemann, who was always available for spontaneous discussions and had an open ear for my problems. He also generated the synthetic data set used in the evaluation, for which I would like to thank him. It has been a great pleasure working in this team.

Furthermore, I like to thank the Institute for Intelligent Cooperating Systems, who let me use their facilities.

I am very thankful for my proofreaders, who pointed out the major and minor flaws in my drafts.

Contents

List of Figures x

List of Tables xi

1 Introduction 1

2 Background 5
2.1 Epidemiology . . . 5
2.1.1 Common Observations . . . 6
2.1.2 Cohort Studies . . . 7
2.1.3 Data Mining in Epidemiology . . . 8
2.2 Classification . . . 8
2.2.1 Fundamentals . . . 9
2.2.2 Classification Models . . . 10
2.3 Terminology . . . 13

3 Requirements 15

4 State of the Art in Model Interpretation 17
4.1 Objectives of Interpretability . . . 18
4.2 Dimensions of Model Explanation . . . 19
4.3 Model-Agnostic Interpretation . . . 20
4.3.1 Feature Importance . . . 21
4.3.2 Feature Grouping . . . 24
4.3.3 Surrogate Models . . . 26
4.4 Interactive Visual Analysis . . . 29
4.5 Summary . . . 33

5 Visual Interface for Model Interpretation 35
5.1 Overview . . . 35
5.2 Data Selection . . . 37
5.3 Model Quality and Feature Importance . . . 39
5.4 Individual Analysis . . . 41

5.5 Partial Dependence . . . 43
5.6 Feature Grouping . . . 45
5.7 Technical Details . . . 46

6 Evaluation **47**
6.1 Prerequisites . . . 47
6.1.1 Methodology . . . 47
6.1.2 Classification . . . 48
6.2 Case Study: Study of Health in Pomerania . . . 50
6.2.1 Study of Health in Pomerania . . . 50
6.2.2 Data Preprocessing . . . 51
6.2.3 Evaluation . . . 53
6.2.4 Comparison with Internal Interpretation . . . 58
6.3 Case Study: Synthetic Data . . . 60
6.3.1 Data Description . . . 60
6.3.2 Evaluation . . . 60
6.3.3 Comparison with Internal Interpretation . . . 62
6.4 Discussion . . . 64

7 Conclusion **67**
7.1 Summary . . . 67
7.2 Future Work . . . 68

A Appendix **71**

Bibliography **73**

List of Figures

2.1 Decision tree 11
2.2 Logistic function 13

4.1 Partial dependence plots 22
(a) Well-performing PDP 22
(b) Poorly performing PDP 22
(c) ICE plot 22
4.2 ExplainVis plot [74] 25
4.3 Illustration of the grouping process in GoldenEye [36]. 25
4.4 Toy example of LIME Ribeiro et al. [72]. 27
4.5 Visualisation of classified instances with ModelTracker [4]. 29
4.6 Explanation of classifying a message as *Hockey* [55]. 30
4.7 Enhanced PDP and PD bars in Prospector [50]. 31
(a) Enhanced PDP 31
(b) PD bars 31
4.8 Visualisation of binary explanations in Krause et al. [51]. 32
4.9 Comparison of classification models via subsets [40]. 32

5.1 Overview of the proposed visual interface. 36
5.2 Data selection panel 37
5.3 Feature importance ranking and comparison. 39
5.4 Individual analysis panel 41

5.5 Partial dependence plot and distribution histogram 44
(a) Continuous feature 44
(b) Categorical feature 44
(c) Continuous feature, all classes 44
(d) Categorical feature, all classes 44
5.6 SilverEye visualisation 45

6.1 Global feature importance scores 54
6.2 Partial dependence plots of `tg_s` and `gg_t` 55
(a) `tg_s` 55
(b) `ggt_s` 55
6.3 Individual analysis of participant 487 56
(a) Class 0 56
(b) Class 2 56
6.4 Comparison of importance scores with and without blood serum biomarkers 57
6.5 Comparison of importance scores with recent and old blood serum biomarkers 57
6.6 Individual analysis of participant 637 58
(a) Old blood serum values 58
(b) Recent blood serum values 58
6.7 Internal interpretation of gradient boosting 59
(a) All features 59
(b) No blood serum biomarkers 59
6.8 Global feature importance scores 61
6.9 Partial dependence plots of `V3` and `V9`. All classes are displayed. . . 61
(a) `V3` 61
(b) `V9` 61
6.10 Comparison of importance scores with and without `V3` 62
6.11 SilverEye scores for `V4` and `V6` 63
6.12 Decision tree trained on the synthetic data set. 63

List of Tables

4.1 Within-class permutations in GoldenEye [35]. 26
(a) Original Data 26
(b) Randomised 26
(c) Randomised 26

6.1 Hyperparameters used in the neural network 49
6.2 Class distributions of $SHIP_M$ and $SHIP_W$ 52
6.3 Classification results on SHIP 54
6.4 Classification Results on the synthetic data 61

A.1 Attributes in SHIP 71

1. Introduction

Epidemiology is concerned with the occurrence and development of diseases in human populations. Epidemiological studies are for example responsible for detecting the relation between smoking and various diseases such as lung cancer and cardiac infarction, which caused a social and political rethinking of smoking. The standard epidemiological workflow for the generation and validation of hypotheses is driven by expert knowledge and statistical methods. With a continuously increasing size of population studies and comprising attributes, the standard workflow becomes time-consuming and is likely to miss relevant information.

A remedy to this challenge may be data mining. Here, the population's data is analysed automatically by machine learning models. Data mining algorithms often perform very well, as they are explicitly designed to handle complex, high-dimensional data. In medical areas such as epidemiology however, it becomes relevant to *understand* the classifier's reasoning behind a prediction. This allows to validate the prediction or find new casual relation. The requirement of interpretability is often neglected in machine learning research, where the primary interest is setting new benchmarks in terms of classification accuracy and solving increasingly complex problems. As a result, classification models become more and more unintuitive, causing a growing gap between predictive power and interpretability.

Thus, techniques are required which explain the reasoning behind a prediction to the user. This demand has recently been recognised by the scientific community, which is reflected by an increasing number in dedicated publications and new summits like the IJCAI 2017 Workshop on xAI[1]. Several political entities accelerate this development, too: The American agency for advanced research projects for

[1] http://home.earthlink.net/~dwaha/research/meetings/ijcai17-xai/, visited 2018-03-15

defense (DARPA) launched a program on explainable artifical intelligence[2], and the European parliament demands a "right to explanation" in automated decision-making[3].

While some literature interprets complex machine learning models for other medical areas such as diabetes diagnosis, no work has been done to support epidemiological research and particularly the analysis of cohort studies. Here, model interpretation suggests to be very promising, as a classifier may reveal meaningful features (or feature combinations) by which an expert can generate or validate hypotheses.

This thesis aims to close the gap between predictive power and interpretability by studying the applicability of *visual model interpretation* for epidemiological data. Visual model interpretation improves the process of understanding predictive models by utilising reasonable visualisations and user interactions, and has been successfully applied in previous work to other medical areas. The contributions of this thesis are as follows:

- Identification of requirements for the interpretation in an epidemiological context. Respective studies share certain characteristics, which need to be specifically considered in the classification and interpretation.
- Review on recent model interpretation techniques with regard to the degree of fulfilment of the identified requirements for epidemiological applications.
- Development of an interactive, visual interface which combines several interpretation techniques. It is suited to analyse classifiers trained on cohort studies by considering the defined requirements. Instead of designing novel interpretation approaches, the capabilities of existing techniques are leveraged and tailored to the specific needs of epidemiological applications.
- Evaluation of the developed interface with respect to the usefulness of its components for understanding the classifier's reasoning. Validation with an internal interpretation.

Thesis Structure

The remainder of this thesis is structured as follows:

Chapter 2 provides background information on epidemiological research, cohort studies and explains the need for data mining in this field. It then gives a basic introduction into classification and presents several classification models.

[2] https://www.darpa.mil/program/explainable-artificial-intelligence, visited 2018-03-15
[3] General Data Protection Regulation, Recital 71

Chapter 3 formulates five requirements, which a model interpretation system in an epidemiological context should satisfy.

In Chapter 4, an overview to the state of the art in model interpretation is given. It consists of a theoretical discourse of the objectives on interpretability and narrows the literature research down to model-independent interpretation techniques. Special attention is given to interactive visual interpretation techniques. The chapter concludes, that the development of a novel interface is required in order to satisfy the requirements of Chapter 3.

A novel interpretation interface is presented in Chapter 5. It explains the single components of the framework along with their aim to improve interpretability.

Chapter 6 evaluates the proposed interface with respect to its effectiveness. The evaluation methodology is explained and classification hyperparameters are presented. Two case studies are performed in order to assess the interface's quality, and a discussion on the results is given.

Chapter 7 summarises the motivation, contributions and insights of this thesis and gives an outlook to potential future work.

2. Background

This chapter aims to make the reader familiar with the background of the thesis. It gives an introduction to the area of epidemiological research, explains the need of data mining in epidemiology, and describes classification as a subfield of machine learning along with several selected types of classification models.

2.1 Epidemiology

The explanations and definitions given in this section are largely based on the descriptions given by Friedman [28] and Preim et al. [70].

Epidemiology is a scientific discipline that studies the "disease occurrence in human populations" [28]. In contrast to clinical medicine, epidemiologists are not concerned with the treatment of a single patient, but capture analyse the causes and effects of health-related conditions in *populations*. Such a population may be for example the citizens of a certain region, or people who are exposed to presumed risk factors of certain diseases. This allows to investigate topics like disease outbreaks and aetiology (i.e. detection of risk factors), health trends, or recommendations for medical check-ups. Despite not directly curing particular patients, epidemiological studies strongly contribute to clinical practices by revealing knowledge about diseases, which can be applied to their diagnosis and treatment.

To understand the relation between a disease and other variables, i.e. potential risk factors, an epidemiologist usually starts by defining a *hypothesis*, which is derived from his expertise or previous studies. The hypothesis is validated by statistical models, which evaluate the effect of the risk factors to the disease's *prevalence* and *incidence*. The prevalence is the portion of people suffering from a disease at a

given point in time; The incidence is the portion of people developing a disease in a given time period. If the expert does not have data about the investigated population at hand, he first needs to assemble it. As it is not feasible to examine the health conditions of all population members, usually a representative, random sample is drawn.

2.1.1 Common Observations

Population studies often gather many different characteristics describing the population members, as this allows to analyse a variety of possible relationships. Some of the most common characteristics are described in the following:

Socio-Demographic Information Socio-demographic information, usually assessed by personal interviews or questionnaires, provides a basic understanding of the participant's background. It not only allows to divide the population into simple subpopulations, but is often relevant factors for disease occurrences. Typical socio-demographic information is the age, sex, marital status, education or income.

Medical Examinations Simple medical examinations are for example somatometric measurements like height and weight, or blood pressure and heart rate. They give a first assessment of the general health condition of the participant, but are somewhat limited in their significance.

Laboratory Data More detailed and reliable information is revealed by laboratory tests. By analysing blood or urine samples for example, experts can determine the glucose level, a known predictor for diabetes.

Medical Imaging Incorporating image data such as MRI or X-ray to an epidemiological study is timely and financially challenging, but provides high-quality information in areas like cardiovascular diseases or liver condition.

Clinical History While the previous observation types measure the participant's health conditions at the time of examination, the clinical history contains much information about previous diseases, diagnoses and treatments, by which the epidemiological expert may explain the current health state.

Of course the observation types used in a study strongly depend on the hypotheses the expert attempts to investigate.

2.1.2 Cohort Studies

This thesis aims to interpret classifiers trained on cohort studies. In the following, a short distinction of different types of epidemiological studies is described, allowing the reader to put cohort studies into context. Then, difficulties in the conduction of cohort studies are discussed.

Study Taxonomy

There are two basic approaches to analyse the behaviour of variables: *Observational studies* do not intervene into the health conditions of the study participants, but let "nature take its course". Changes in a variable can then be explained by changes in other variables. *Experimental studies* on the other hand actively intervene. Here, the experimenter may for example give a drug to a portion of study participants and observe its influence.

Observational studies can be divided into descriptive and analytic studies, which again can be subdivided into prevalence, case-control and cohort studies. A *descriptive study* performs, as the name already suggests, only a description of occurring diseases or disease-related phenomena. It gathers information of study participants by conducting tests (such as those described in the previous section) and summarises it into a database. Descriptive studies do not attempt to directly validate hypotheses, but give potential to generate new hypotheses. In contrast, *analytic studies* test existing hypotheses and therefore seek to *explain* a disease.

Prevalence (or *cross-sectional*) studies analyse the relation between a disease and other given variables at one particular moment in time in a predefined population. *Case-control* studies differ from prevalence studies in the way that they study the relationship of an *existing* disease and other variables; Therefore, case-control studies observe only subjects with the disease-of-interest. *Cohort studies* follow a population over a particular time. This allows to monitor the development of diseases. Here, the same participants are examined multiple times over the time period of observation.

Challenges

The analysis of cohort studies in order to understand the development of diseases and temporal influences is promising. However, the conduction process comes with several challenges, which have an impact on the data quality: (1) Not all participants are tested on the same variables. This may have different reasons: Some individuals for example can not be recorded in an MRI, as they have tattoos, dental braces or metallic implants. Others simply decline some examination types. Some variables are sex-specific: Women may be asked about menstrual information

and childbirth status; Men may be tested towards erectile functionality. (2) Not each examination moment in a cohort study necessarily contains the same attributes. Some instruments such as MRI or ultrasound may be added in a later moment, or removed at some point. (3): Often, participants drop out of the observed population, because they move, decease, or do not react to the epidemiologists' invitations any more. (4): In interviews and questionnaires, participants tend to lie, if they are ashamed of their health and social circumstances (e.g. alcohol or drug use). To guarantee a sufficient degree of statistical reliability, cohort studies are therefore required to start with a large population sample.

2.1.3 Data Mining in Epidemiology

In their "call for biological data mining approaches in epidemiology", Lynch and Moore [61] explain that traditional epidemiology focuses on univariate analysis and validation of simple hypotheses, which consider only a small number of risk factors. Recently, due to the complex nature of many diseases, new population studies grow in size and number of captured attributes. Standard epidemiological workflows can often not cope any more with the data volume and possibilities of existing risk factors. Epidemiology is therefore in the need of "more powerful modeling approaches" [61], which are able to handle large-scale, heterogeneous data.

Several publications follow this argumentation and apply data mining to epidemiology. Buczak et al. [17] for example predict dengue fever with fuzzy association rule mining; Li et al. [58] use decision trees and classification rules to detect risk patterns in medical data sets. They argue their choice of classification method with its interpretability: "In general, medical practitioners and researchers do not care how sophisticated a data mining method is, but they do care how understandable its results are".

Many relevant publications use ruled-based algorithms or decision trees, likely due to their interpretability. Other model types such as neural networks or support vector machines are rarely used, despite their predictive performance.

2.2 Classification

Conventional hypotheses-driven epidemiological studies are challenged by increasing data sizes, which is why recently data mining approaches are incorporated into the evaluation of population studies. This section explains the fundamentals of classification, a discipline in data mining. Then, it presents a selection of classification model types, which will be used in the evaluation.

2.2.1 Fundamentals

The following explanations are based on Tan et al. [82], if not specified otherwise.

Let X be a set of instances, each described by a set of attributes $\mathcal{A}$, and C be a set of discrete and mutually exclusive classes. Then classification denotes the task of learning a function f that maps an instance in X to an element of C[1]:

$$f : X \mapsto C \tag{2.1}$$

Such a function is usually called *classification model* (or *classifier*). Many approaches have been developed to automatically determine a classification model, usually by employing a learning algorithm. Such an algorithm takes as its input a set of instances with known class assignment, and attempts to identify a model which fits the relationship between input data and class attribute best. A key objective of a classification model is generality: Not only is the model supposed to correctly classify instances of the input set, but also new, unknown instances. A model which is trained to only predict the instances of the input set well is said to be *overfitted*.

Evaluating the Performance

To test the generality of a classifier, the input set is usually divided into a *training set* and a *test set*. A classification model is then developed based only on the training set. Afterwards, its quality is analysed by classifying the instances of the test set and comparing the predicted classes with the true class information. Here, the most popular evaluation metric is the *accuracy*. It is computed as the ratio of correctly classified instances and the size of the test set:

$$\text{Accuracy}(X) = \frac{1}{|X|}|\{x|x_Y = x_C, x \in X\}| \tag{2.2}$$

where x_Y and x_C are the predicted and true class of instance x, respectively. While the model accuracy is simple to determine and very popular in the literature, it is prone to imbalanced data, i.e. data sets where the classes are not evenly distributed [54]. As an example imagine a data set, of which 98% of the instances belong to the negative class, and only 2% belong to the positive class. A classifier always predicting the negative class would achieve a very high accuracy of 0.98. Such an

[1] This notation will be used consistently throughout this thesis, and extended where necessary.

imbalance is not unusual for medical data, as even common diseases are rare when compared to the population's health.

McHugh [65] advises researchers in health-related areas to additionally measure the kappa statistic (or *Cohen's kappa*), which takes class imbalances into account. Kappa quantifies the agreement between two raters (e.g. classifiers) into the interval [-1, 1], where a value of 0 denotes the amount of agreement that can be expected from random chance, and a value of 1 denotes perfect agreement. Values below 0 indicate agreement even worse than expected. It is commonly used to compare a classifier to the ground truth. Kappa compares the *observed agreement* (i.e. accuracy) to the *expected agreement*:

$$\text{ExpAgreement}(X) = \frac{1}{|X|^2} \sum_{c \in C} | \underbrace{\{x | x_Y = c, x \in X\}}_{\text{instances predicted as } c} || \underbrace{\{x | x_C = c, x \in X\}}_{\text{instances belonging to } c} | \tag{2.3}$$

$$\text{Kappa}(X) = \frac{\text{ObsAgreement}(X) - \text{ExpAgreement}(X)}{1 - \text{ExpAgreement}(X)} \tag{2.4}$$

Probabilistic Classification

In contrast to standard classification, where the classification task is to predict the class of an instance, in *probabilistic classification* the model computes the probability of an instance belonging to class c:

$$f_c : X \mapsto [0, 1] \qquad \text{with} \sum_{c \in C} f_c(x) = 1 \tag{2.5}$$

Probabilistic predictions potentially offer more information about an instance, as they allow for a more detailed comparison. Imagine two instances, which are classified by a non-probabilistic and a probabilistic classifier. The non-probabilistic classifier predicts that both instances belong to the same class. The probabilistic model however additionally estimates that the first instance belongs to the predicted class with a probability of 90%, while the second instance belongs to the predicted class with a probability of 60%. Hence, the classifier is more certain about the first instance than about the second instance.

2.2.2 Classification Models

This section introduces five types of classification models. The selection of presented types is based on their interpretability: Four of the five described types are well-interpretable or already provide metrics for interpretation. This appears to be

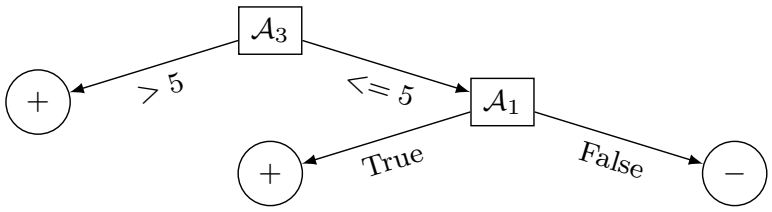


Figure 2.1: Simple decision tree

in contradiction with the objective of explaining unintuitive models. However, it allows to compare the techniques, which will be used in the developed framework, with model-specific interpretations. The fifth selected model type – neural networks – are difficult to interpret without proper techniques and are therefore a reasonable representative of a true "black-box" model.

Decision Tree

A decision tree is a simple and popular model type for both classification and regression of instances with numerical, categorical or mixed-type attributes. A decision tree is a tree-like graph, where each node represents a question (decision) with respect to the value of an attribute, and each edge represents a possible outcome of this question. An exemplary decision could be: *Has the patient diabetes?* with possible outcomes *Yes* and *No*. Each leaf of the tree represents a prediction with respect to the target variable. An instance is therefore classified by – starting at the root node – following the decisions, until it arrives at a leaf, where a class is assigned to the instance. Figure 2.1 shows an exemplary decision tree.

Decision trees are induced by a simple, iterative strategy: In each step, the induction algorithm iterates through the attributes and estimates, how much each attribute reduces the *impurity* with respect to the class attribute. The attribute with the highest impurity reduction (*gain*) is selected as the next node. Continuous attributes need to be discretised into intervals before, either manually by the user or automatically.

Random Forest

While decision trees are a simple, intuitive way to classify instances, they have several disadvantages. If they do not use any regularisation, they tend to become very deep, resulting in overfitting caused by high variance [34]. This problem has been addressed by Breiman [14], who developed bootstrap aggregating (bagging). Here, instead of a single classifier, an ensemble of classifiers is learned, each on a

different data subset. An instance is predicted as the class which the majority of ensemble members supports. Often, decision trees are used as the base classifiers. Although bagging reduces the variance, it suffers from correlation between the classifiers [34]. Random forests reduce the correlation between the underlying decision trees by taking a random selection on the features for each bootstrap sample, resulting in a better variance reduction.

Gradient Boosting

Another type of ensemble learning is gradient boosting. In contrast to bagging and random forests, the classifiers are not learned independently and simultaneously. Instead, a stagewise additive model is learned, i.e. a model created by successively incorporating "weak" classifiers. Again, decision trees are a common choice as the underlying classifiers. In each iteration, a tree is added to the model such that a loss function, for example the mean squared error, is minimised [29]. To avoid overfitting, a regularisation term is usually used.

Logistic Regression

Logistic regression is, despite its name, not only a regression model, but can be used for classification. Its goal is the same as that of other regressions techniques: Finding a function that represents and generalises the data. In contrast to other techniques, the target variable is restricted to be binary or dichotomous [39]. The idea of logistic regression is to express the relationship between a set of numerical attributes and the class attribute by the logistic function (Equation 2.6 and Figure 2.2). The result of the logistic function can be interpreted as the probability of belonging to the positive class.

Learning a logistic regression model means to estimate the coefficients $\beta_0, \beta_1 \dots \beta_n$, usually via maximum likelihood estimation [39]. Here, a *logit transformation* is performed first, which allows to turn the optimisation into linear form (Equation 2.7).

$$f(t) = \frac{e^t}{1+e^t} \qquad \text{with } t = \beta_0 + \sum_{i=1}^{|X|} \beta_i X_i \tag{2.6}$$

$$g(x) = \ln\left(\frac{f(x)}{1-f(x)}\right) \tag{2.7}$$

The formula and descriptions above are related to the standard, binary classification task. Logistic regression however can also be extended to fit one or multiple logistic functions to classify data with more than two classes [39]. Due to its simple mathematical formulation and potential for interpretation, logistic regression is popular among scientists in various fields, including epidemiology [48].

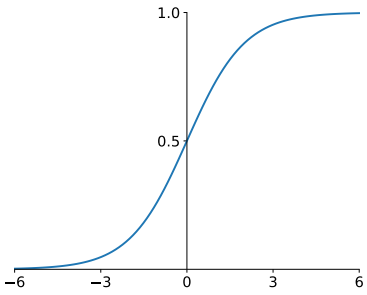


Figure 2.2: Logistic function

Neural Network

Neural networks are inspired by the human nervous system. They consist of several units, the neurons, which are grouped into layers. A neuron processes information with an activation function on their input and translates the output to the neurons of the next layer. The output of the last layer then determines the classification. A network consists of at least two layers: One input layer and one output layer, with potentially several so-called hidden layers in between them. To fit a neural network to a classification problem, the weights of the connections between the neurons are adjusted. This is done by gradually propagating the error, i.e. the difference between output and expected output, back through the network [53].

2.3 Terminology

The thesis at hand covers aspects of epidemiology, machine learning, statistics and visualisation. This interdisciplinarity comes with a pitfall: While studying related publications of the different scientific communities, as well as in discussion with experts, it became apparent that there is sometimes no agreement on terminology. Sometimes, communities use different terms, which have essentially the same meaning, or where the difference is not relevant in the context of this thesis. On the other hand, some terms are shared by several communities, but differ in their meaning depending on their context. In the following, some of the most important terms are defined and explained for the given context. Note, that this glossary does not attempt to be complete with respect to the technical vocabulary used in this study, but is only concerned with terms that may cause misunderstanding or confusion to some readers.

Prediction and Classification

The definition of a prediction is controversial. Generally, it denotes the assignment of a missing value to an instance, based on domain knowledge (e.g. a learned

model). Some authors distinguish a prediction from a classification in terms of the target attribute: Classification assigns *categorical* values, prediction *continuous* values [33]. Other authors use prediction in the context of time series and streams, where future events and conditions are estimated, such as in weather forecasting [5]. This thesis however is neither concerned with future events nor regression tasks, but only classification. Thus, the terms prediction and classification will be used interchangeably, if no other context is specified.

Attribute and Target Attribute

An attribute is a describing property of an instance. In an epidemiological study for instance, each study participant is characterised by various properties such as age, sex or blood pressure. In machine learning, attributes are also denoted as feature, predictor or simply variable. In the context of regression, attributes are usually called covariates, explanatory variables or independent variables [39].

In a prediction task, each instance is assigned a value, which belongs to the target attribute. Again, different terms can be found, such as class, label or output attribute. In regression, the term dependent variable is popular.

Binary, Multi-Class and Multi-Label Classification

The simplest scenario in classification is a binary classification. Here, the set of classes only consists of two elements, often denoted as the *positive* and *negative* class. A generalisation is multi-class classification. Here, the target variable may contain an arbitrary number of elements. Another variant is multi-label classification, which drops the constraint of labels being mutually exclusive. Hence, an instance may be assigned to multiple classes.

Interaction

Epidemiological studies often analyse interactions. Here, an interaction exists, if a variable's value may "depend in some way on the presence or absence of another [variable]" [86]. In computer science, the term interaction is often used on the context of human-computer interaction or interactive systems. The different background of this term has previously led to some confusion in discussions[2].

A clear explanation of the term seems to be necessary for this thesis, as it is concerned with both contexts: The proposed *interactive* visual interface aims to support epidemiological experts in finding feature *interactions*. In the following, the term interaction is used to describe human involvement in a visual interface. The dependence between variables is instead denoted as *feature combinations*.

[2]Bernhard Preim, personal communication, 2017-10-26.

3. Requirements

The interpretation of classification models learned on epidemiological data requires a consideration of several characteristics. Before reviewing the related work on model interpretation, it is necessary to make oneself familiar with these requirements in order to discuss properly, how existing approaches can help.

Requirement 1: *Handle multivariate, heterogeneous data sets.*
Epidemiological studies contain information on the background and the health conditions of their participants. Study members may be characterised by various factors, such as socio-demographic factors, medical examinations or laboratory data. For a single individual, dozens or even hundreds of features may be collected. Hence, an analytics tool working on such data needs to be reasonably scalable to a high number of attributes per instance. Furthermore, the collected features may contain binary (e.g. sex), real-valued (e.g. age) or categorical (e.g. education) information.

Requirement 2: *Support longitudinal features.*
In cohort studies, participants are repeatedly examined over time, in order to observe their development. This results in multiple values per attribute, one for each examination moment. This information needs to be considered, especially in distinction to values of other attributes.

Requirement 3: *Be independent in the choice of the used classification model.*
Data-driven methods for epidemiology use a variety of machine learning models, such as decision trees, neural networks or support vector machines [42]. To make an interpretation tool applicable to many epidemiological experts, it should not be restricted to a certain type of classification model.

Requirement 4: *Allow multi-class classification.*
Epidemiological studies are not restricted to binary classification, but often investigate problems with multiple outcomes, e.g. diabetes type detection [42].

Requirement 5: *Provide a visual model interpretation system.*
Krause et al. [49] plead for using visual analytics in model interpretability for two reasons: (1) Humans are better than machines in solving some tasks, and (2) human understanding and interpretation is desired. A more theoretic argumentation is given by Weld and Bansal [88]. They encourage the development of interactive explanation systems, as they allow follow-up questions and actions of the user and thus enable a "dialogue" between the user and explainer. This suggestion is supported by research in psychology, which shows that an explanation is most effective in a conversation between explainer and explainee [66].

4. State of the Art in Model Interpretation

Machine learning has recently led to dramatic success in various applications such as image classification [52], epidemiology [68], or winning complex board games against human champions [76]. With over 500 journals and yearly conferences[1], it is one of the most dynamic research areas not only in classical computer science, but also many applied disciplines benefit from analysing large amounts of data.

Especially in those applied areas, a key requirement is being able to interpret the data mining system. A physician, whose machine learning model has classified a patient to suffer from cancer, needs to scrutinise and justify this diagnosis, before any treatment can be started. The ability to explain a prediction is therefore a highly desired feature for a decision-assisting system [13]. Unfortunately, this aspect is often neglected in new advances in machine learning, as the primary interest is only to push towards a higher accuracy. New state-of-the-art models become hard to interpret properly, causing a gap between prediction accuracy and interpretability [54].

By now, the need for explainable artificial intelligence has been acknowledged by the scientific community, and an increasing number of research groups dedicate their work to close the gap between accuracy and interpretation in machine learning. This chapter aims to give an insight into the different aspects of model explanation and discusses, which techniques are appropriate for the topic of this work, and which are not. Due to the extensive literature on the investigated topic, this overview does not attempt to be complete, but only highlights the most relevant work. For

[1] http://www.scimagojr.com/journalrank.php?category=1702, visited 2018-03-13

a comprehensive survey on model interpretation, the reader may be referred to Guidotti et al. [32].

4.1 Objectives of Interpretability

Before an overview of existing work in the area of model interpretation is presented, it may be worth to clarify what interpretability in the context of data mining means and what its objectives are. Doshi-Velez and Kim [24] define interpretability as the "ability to explain or to present [a machine learning system] in understandable terms to a human", and state that it is used to confirm other desiderata. A list of such desiderata is given by Lipton [59]:

Trust Trust is often defined as the confidence that the model performs well. The performance of a model is usually quantified on hold-out data tested on the learned classifier. However, the resulting measures are prone to the data being biased. To build trust into a system, it might be therefore interesting, too, to understand why examples are classified correctly or why certain mistakes are made.

Informativeness The purpose of using data mining is to gain useful information about instances. While this is mainly done by computing a prediction, other information may be equally relevant, such as (not) similar instances, outliers etc. As such information is often not provided by the model itself, it can be mined by interpreting the model.

Causality Machine learning models do not find causal associations, but only correlations and data-based associations. However, by interpreting a model, domain experts may generate or find evidence for a hypothesis. Here, the motivation is to use the ability of classifiers of finding complex feature interactions that are hard to find in the data itself. Supporting epidemiologists in inferring and validating causal relationships in cohort studies is the motivation of this thesis, which is why causality may be seen as the main desideratum in this work.

Fair and Ethical Decision-Making Of special interest for consumer advocates, politicians and ethicists is to ensure that automated decision-making follows ethical standards. Model interpretation can for example help to understand, if the classifier has any bias towards any ethnicity or gender.

Transferability In real-world applications, classification models are learned with the purpose of being deployed later into a productive system, where it has to

classify new, unknown instances. If the learning data is biased, or the real-world data alters, the classifier may lose its predictive power. Transferability ensures that the model is robust and general enough to withstand a slightly changing environment. Model interpretability can help to investigate whether the classifier has achieved the desired generalisation or not.

Note that this list of desiderata is not complete. Certain applications may have different or additional objectives, for example privacy in a data protection setting. Generally, it can be stated that interpretability is needed wherever the sole prediction is incomplete for satisfying the objective, i.e. where a gap between the formal error optimisation and the original application purpose exists.

4.2 Dimensions of Model Explanation

Existing approaches for the explanation of classification differ in several dimensions: Some are only designed for specific model classes or data types, some require the model to be already learned, etc. This section introduces the different categories of model explanation and discusses, if they are applicable for the scope of this thesis.

Transparency vs. Post-Hoc Explanations

Lipton [59] distinguishes techniques by whether they try to improve the transparency of a prediction model, or give post-hoc explanations. Here, transparency is considered as the understanding of how the mechanism behind a model works. Such transparency can be achieved on different levels: The whole model can be understood (*simulatability*), single components like the neurons in a neural network can be understood (*decomposability*), or at least the learning strategy can be understood (*algorithmic transparency*). Post-hoc explanations on the other hand extract, as their name already suggests, information from an already-learned model. They do not elucidate how the model mechanism works, but instead look for other useful information. Post-hoc explanations can for example compute the importance of a feature for the prediction, deliver similar instances for a given prediction, or visualise the classification space and relevant areas therein.

Model-Gnosticism

Another way of classifying model interpretation techniques is to differentiate, whether they have knowledge about the internal algorithmics of a model or not (*model-gnostic* vs. *model-agnostic* explanation). A model-gnostic explanation has a significant advantage over a model-agnostic explanation, as additional knowledge about the prediction process is available. A popular example for model-gnostic

explanation is the *Neural Interpretation Diagram*, a visualisation of neural networks, where the edge thickness and shading depends on the edge weights [69]. In an agnostic setting, the model is treated as a black-box, where only input and output can be observed. Any information about the model can therefore only be obtained by interpreting the changes in the output for certain input. This comes with a benefit: Model-agnostic approaches can be easily added to, exchanged in, and removed from a data mining pipeline, as they can be treated like an abstract module. A common example of model-agnostic techniques is to quantify the importance of features to the prediction.

These two ways of dividing interpretation techniques into groups correlate strongly with each other. Giving transparency to a model requires knowledge about its mechanism; Techniques aiming for transparency are therefore model-gnostic. For the same reason, model-agnostic strategies can only give post-hoc explanations, as they are only able to observe the output of a learned model and cannot look into it. On the other hand, model-gnostic approaches can also aim for post-hoc explanations and vice versa.

Explanation Scope

Another dimension to be considered is the scope of explanation. A method may either attempt to explain the whole model, allowing the interpretation of all inputs and outputs. This is called *global interpretability*. In other cases, a method explains the logic behind the prediction of a single instance, i.e. allows for a *local* (or *individual*) interpretation.

Data Type

Some methods target specific types of data. Several publications investigate for instance, which regions in an image are decisive for a prediction [22], others are concerned with text mining [63]. For tabular data, methods may be restricted to binary data, while others support both continuous and categorical data.

4.3 Model-Agnostic Interpretation

This thesis does not intend to restrict epidemiological experts in their choice of the used data mining technique, as stated in Requirement 3. Instead, it aims to support an expert in understanding the prediction model's outcomes, regardless of the chosen model type. Hence, the literature review will focus on *model-agnostic* interpretations. Furthermore, Requirement 1 demands the capability of handling heterogeneous data, as cohort studies contain various attributes. For simplicity, it is assumed that this information is processed to raw, tabular data. Hence, publications

only targeting image or textual data are not introduced. Requirement 4 asks for non-binary classification: Some of the methods in the following overview are originally applied to binary classification, but can be adjusted to the multi-class setting. Section 4.4 addresses interactive frameworks, as stated in Requirement 5.

The overview is divided into four categories, depending on the underlying fundamental idea of explanation. *Feature importance* methods analyse, how a single feature contributes to the prediction(s). Another concept is to find the optimal *feature grouping* in a data set. Other approaches again explain a classification or classification space by *surrogate models*, often rules or decision trees. Few publications consider incorporating the user into the analysis process by suggesting *interactive visual analysis*.

4.3.1 Feature Importance

One of the most popular ideas to gain understanding of a classification model is to analyse how important a feature is, i.e. how much the model relies on its value during the prediction process. This information is highly relevant in many applications: Medical experts may detect critical health risk factors; Advertisement companies may conclude, which information about consumers explains their shopping behaviour best; Car salesmen may understand, which car property is the most critical to the customers, etc.

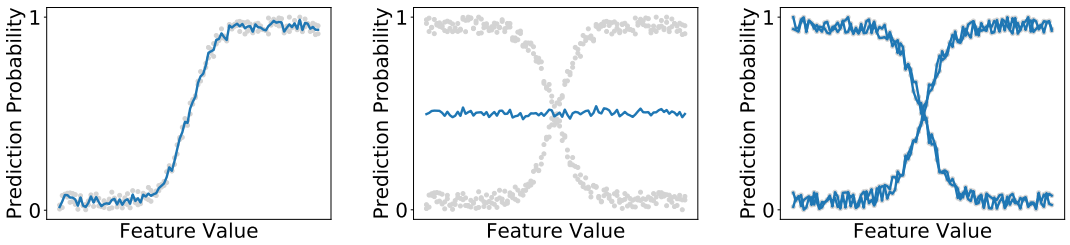
Partial Dependence

The relationship between the value of an attribute (or subset of attributes) and the target value, also known as partial dependence (PD), has been first computed and visualised by Friedman [29]. Let S be a subset of the attributes in a dataset, $S \subseteq \mathcal{A}$, and let $R = \mathcal{A} \setminus S$ be the complementary subset, i.e. the remaining attributes. Moreover, let f be a binary, probabilistic classifier. The partial dependence of subset S is the *expectation* $\mathbb{E}$ over the marginal distribution of the complement subset R. In other words, it is the prediction probability of an instance x with a fixed value $S = s$, integrated over all possible values of R:

$$\mathrm{PD}_S(s) = \mathbb{E}_R\left[f(x \leftarrow S = s)\right] = \int f(x \leftarrow S = s) dR \tag{4.1}$$

As iterating over the marginal distribution is not computable in reasonable time, a Monte-Carlo approximation is performed, where the PD value is estimated as the average over all instances in the data:

$$\hat{\mathrm{PD}}_S(s) = \frac{1}{|X|} \sum_{x \in X} f(x \leftarrow S = s) \tag{4.2}$$



(a) PDP **(b)** PDP **(c)** ICE plot

Figure 4.1: Partial dependence plots. Grey dots represent the classifier results. (a): The PD correctly captures the relation between feature value and prediction probability. In (b), two subpopulations exist, but are averaged out in the PD. (c): An ICE plot preserves the subpopulations.

The values s are determined by sampling from the value range of S. The results are visualised in a line chart called partial dependence plot (PDP). Two examples are shown in Figures 4.1(a) and 4.1(b). Originally applied to gradient boosting classifiers, the concept of a PDP has been adopted to various other classification techniques, most famously random forests [26].

Goldstein et al. [31] argue that a PDP does not work well if the investigated variable contains subpopulations, as the averaging removes such information (see Figure 4.1(b)). Their extension, individual conditional expectation (ICE) plots, replaces the average partial response curve by one curve for each input (see Figure 4.1(c)). As pointed out by Krause et al. [49, 50], ICE plots are prone to visual clutter. For the basic PD, they present an improved sampling technique which takes the feature value distribution into account. Apley [7] proposes another type of visualisation called accumulated local effect (ALE) plot, which is less computationally intensive than a PD plot and avoids the problem of depending variables.

Line charts like PDP, ICE or ALE all share a major disadvantage: They scale badly, if one wants to investigate not only the impact of one feature, but multiple features simultaneously. While it is still possible to display the interaction of two features in a three-dimensional surface plot, the visualisation of more than two features becomes difficult and unintuitive. Nonetheless, partial dependence plots are very common for a first analysis of a classifier's reasoning and often found in implementations and scientific studies.

Global Importance Values

The aforementioned approaches are qualitative measures for the importance of a feature; The interpretation of the resulting curves is left to the analyst. If the feature

importance computation is part of an automated analytics pipeline and needed for subsequent stages like feature selection, it may however be necessary to provide a measure that quantifies the global impact of a variable. A possible solution for this problem is a *sensitivity analysis*, which studies the relation between the input's uncertainty and the uncertainty in the classification output [32, 75]. Here, the classifier's response to a varying input (e.g. the partial dependence) is aggregated to a single value by applying a sensitivity measure, for example the output range or variance. Sensitivity analysis has been used to explain black-box models by Cortez and Embrechts [19, 20], but is also applied to feature selection in an iterative optimiser [27][2]. Lemaire et al. [57] state that sensitivity analysis is misleading, if the sensitivity response is not monotonous. They propose an importance measure that takes the integral of the response function and the empirical probability distribution of the variable into account.

Other studies criticise that only the direct feature influence is measured, but not the indirect influence, which additionally indicates the dependence between features [1, 79, 89]. As an example, one may consider a classifier that decides on granting a housing loan or not. While a direct feature importance measure could suggest that an applicant's race is not relevant to a decision, an indirect importance measure may reveal that the property's zipcode correlates with the race. To compute this influence, one can either iterate over all feature combinations (marginalise) [79, 89] or impute a feature from the remaining features [1].

Local Importance Values

Most of the publications described above address the overall importance of attributes in the model's decision process. For an expert it is however often equally relevant to understand, which features contribute to the prediction for a single instance, e.g. a patient. Robnik-Šikonja and Kononenko [74] observe how a binary prediction probability changes if a feature $\mathcal{A}_i$ is ignored by the classifier. First, the "absence" of the feature $x \setminus \mathcal{A}_i$ is approximated by cloning the inspected instance, each with a different value of the investigated feature, and weighting the prediction results with the probability of the replacement value:

$$f(x \setminus \mathcal{A}_i) = \sum_{a \in \mathcal{A}_i} f(x \leftarrow \mathcal{A}_i = a) P(\mathcal{A}_i = a) \tag{4.3}$$

[2]Feature selection aims to reduce computational costs and improve the classifier quality by removing features from the data *before* the learning phase, usually by detecting highly correlated attributes or attributes that do not explain much variance. In contrast, feature importance analyses, which features are relevant to the classifier *after* it has been trained.

Numerical values can not be directly processed by this technique, but are discretised into intervals beforehand. After determining the "ignoring effect", the *prediction difference* between original probability and ignoring probability is computed. The authors propose three different ways of computing this difference: The direct probability difference (Equation 4.4), the difference in information (Equation 4.5), or the weight of evidence (Equation 4.6). Thus, the maximum difference depends on the original prediction probability.

$$\text{predDiff}_i(x) = f(x) - f(x \setminus \mathcal{A}_i) \tag{4.4}$$
$$\text{predDiff}_i(x) = \log_2 f(x) - \log_2 f(x \setminus \mathcal{A}_i) \tag{4.5}$$
$$\text{predDiff}_i(x) = \log_2(\text{odds}(x)) - \log_2(\text{odds}(x \setminus \mathcal{A}_i)) \tag{4.6}$$
$$\text{with} \quad \text{odds}(x) = f(x)/(1 - f(x))$$

The resulting importance values are displayed in a horizontal bar chart called *explainVis*, as shown in Figure 4.2. In a follow-up study, Štrumbelj et al. [79] address the indirect influence of a feature by marginalising it, i.e. iterating over the power set of all feature combinations. As the computational costs grow exponentially with an increasing number of attributes, reasonable sampling and approximation schemes become necessary [77, 78].

Another way of formalising the feature importance is from a game-theoretical point of view. Here, the Shapley value determines how much a player contributed to a coalition and assigns his share of the game's output. The contribution of a player is defined as the difference in the game output with and without the player. Again, the challenge is to find a good approximation. Štrumbelj and Kononenko [77] estimate the Shapley value using Monte-Carlo simulation; Lundberg and Lee [60] use weight kernels and linear regression.

4.3.2 Feature Grouping

Henelius et al. [35] aim to find the optimal feature grouping of a learned classifier as an approach to detect meaningful feature combinations. Starting with a group containing all features, their algorithm *GoldenEye* iteratively constructs a solution tree by removing one feature at a time from the group and measuring the impact to the prediction quality. If the quality drops below a threshold, the tree branch is exhausted. The final group is the best group among the leaves. An exemplary grouping process is shown in Figure 4.3. Once the final group is discovered, the next best group is determined from the remaining attributes. The prediction quality of a grouping is computed by within-class random permutations: All features of

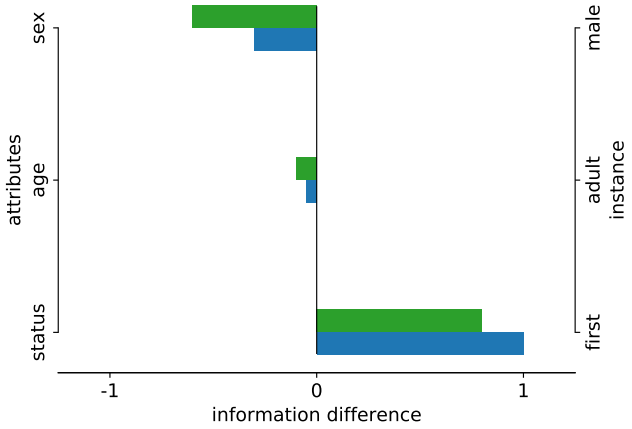


Figure 4.2: ExplainVis plot [74] of a classifier predicting the survival of a Titanic passenger. Blue bars represent the individual feature importance, green bars the average feature importance. The left axis spine displays all available feature; The right spine shows the corresponding values for the individual. A bar in right direction indicates a feature's contribution towards a survival, and vice versa.

the same group are permuted mutually with respect to the predicted class. The effect of the grouping G is then characterised as the fidelity, which is the fraction of matching predictions between the original (unrandomised) data set X and the randomised data set X^*_G:

$$\text{fidelity}(X, X^*_G) = \frac{1}{|X|}|\{i|f(x_i) = f(x^*_{G,i}), i = 1, \ldots, |X|\}| \tag{4.7}$$

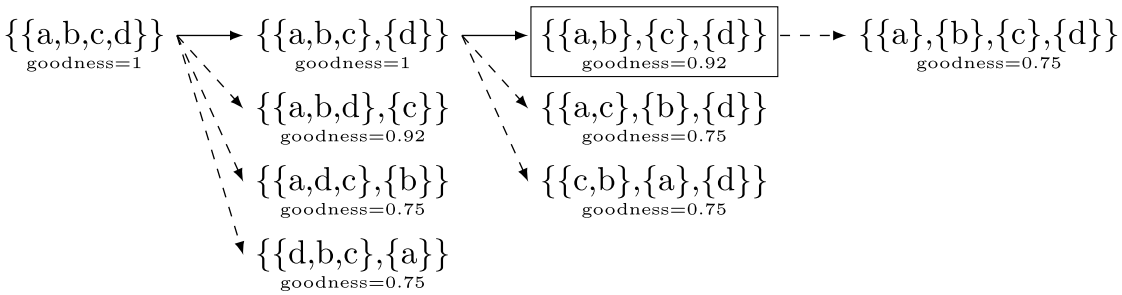


Figure 4.3: Illustration of the grouping process in GoldenEye. For the threshold $\Delta = 0.8$, the best group is {a,b}. Visualisation inspired by Henelius et al. [36].

Table 4.1: Within-class permutations. The data set is characterised by the attributes $\mathcal{A} = \{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3\}$ and class attribute c. (a): the classifier has correctly learned the relation $y = f(\mathcal{A}) = \mathcal{A}_1$ XOR $\mathcal{A}_2$. (b): $\mathcal{A}_1$ and $\mathcal{A}_2$ have been permuted independently, causing mistakes in the new predictions y^* (circled instances). (c): permuting the attributes together preserves their relationship.

(a) Original Data

c	y	$\mathcal{A}_1$	$\mathcal{A}_2$	$\mathcal{A}_3$
−	−	0	0	0
−	−	0	0	1
−	−	1	1	0
−	−	1	1	1
+	+	0	1	0
+	+	0	1	1
+	+	1	0	0
+	+	1	0	1

(b) Randomised

y	y^*	$\mathcal{A}_1$	$\mathcal{A}_2$	$\mathcal{A}_3$
−	⊕	1	0	0
−	⊕	0	1	1
−	−	0	0	0
−	−	1	1	1
+	+	0	1	0
+	⊖	1	1	1
+	+	1	0	0
+	⊖	0	0	1

(c) Randomised

y	y^*	$\mathcal{A}_1$	$\mathcal{A}_2$	$\mathcal{A}_3$
−	−	1	1	0
−	−	0	0	1
−	−	1	1	0
−	−	0	0	1
+	+	0	1	0
+	+	1	0	1
+	+	1	0	0
+	+	0	1	1

An example of within-class permutations is shown in Table 4.1. In a follow-up work, Henelius et al. [36] replace the fidelity measure by correlation goodness, as fidelity is susceptible to class imbalance. Here, correlation goodness is the "similarity between the predicted class membership probabilities of the original and the randomized datasets". No other work has been found that specifically searches for meaningful feature combinations.

4.3.3 Surrogate Models

Some authors propose to reproduce the behaviour of a non-interpretable classification model by another, interpretable model or representation. These alternative representations are called *surrogate models*.

Local Interpretable Model-Agnostic Explanations

Ribeiro et al. [72] introduce local interpretable model-agnostic explanations (*LIME*). Their assumption is that any prediction space can be locally explained by a linear model. For a given instance, several perturbed instances are generated (e.g. by greying out some parts of an image) and weighted by their similarity to the original instance. Then, a simple interpretable model, for example a linear regression, is learned on the perturbed instances. This model explains the original instance. Figure 4.4 shows a toy example for such local interpretation. To gain a global

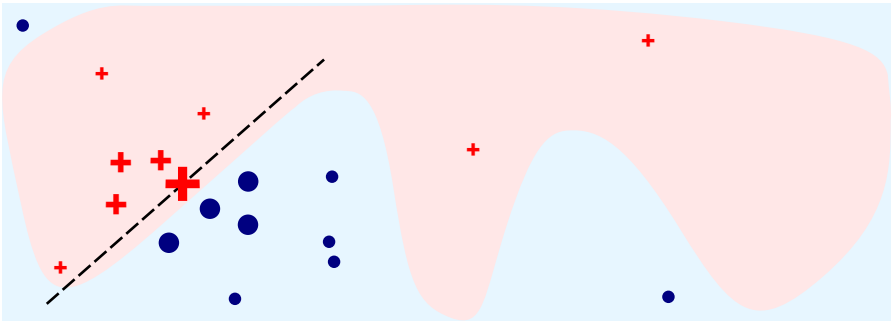


Figure 4.4: Toy example of LIME. The red (blue) area represents the space of the positive (negative) class. An instance, here the bold red cross, is locally explained by sampling similar instances and weighting them by similarity (here represented by size). The dashed line represents the locally learned model. Visualisation inspired by Ribeiro et al. [72].

understanding of the model, a set of diverse and non-redundant explanations can be picked. LIME has gained much popularity and is implemented in many programming languages and analytics tools. Ribeiro et al. [73] point out that the explanations generated by LIME and other local approaches do not sufficiently inform about their generality, i.e. whether they also apply to new, unseen instances. In their work, they developed an algorithm that produces *anchors*. Anchors are explanations sufficient enough for the prediction. In other words, any instance for which the anchor holds will be predicted equally.

Rule Extraction

Another common idea is to extract decision rules from a trained classification model. This approach has become popular especially for specific model types such as neural networks [6] or support vector machines [62]. A model-agnostic strategy has been proposed by Kim and Seo [46]. They create a contribution matrix containing the individual feature contributions by applying LIME. By applying a non-negative matrix factorisation, the resulting column vector then contains information about groups of features that simultaneously affect the prediction, i.e. rules. Bastani et al. [9] approximate any black-box model by a decision tree – a well-interpretable model. For the extraction, a sample of instances labelled by the trained classifier is used.

Other work addresses the extraction of rules for a single instance. Turner [85] designed a model explanation system that assigns a score to an explanation and uses Monte-Carlo to find an approximately optimal solution. Tamagnini et al. [81] find explanations for binary-encoded instances by iteratively "removing" one

feature at a time, until the prediction changes. Here, "removing" is defined as changing the value of a present feature to not present. This obviously only works for binary-encoded data. The set of remaining features is then an explanation. The explanations are visualised in an interactive interface, allowing for more detailed information.

Case-Based Explanations

Commercial applications often classify new instances by comparing it to similar, already classified instances. This method, called case-based reasoning (CBR), can be extended to elucidate the reasoning behind a prediction by naming similar cases and their difference to the new instance. In their study, Cunningham et al. [21] conclude that CBR is more convincing than rule-based explanations. Kim et al. [44] propose a variant of a Bayesian network that explains predictions with cluster prototypes. However, these approaches are not model-agnostic. To apply the idea of case-based explanations to any prediction model, one has to find representative samples of the prediction space. This has been solved in a cluster-based strategy by Bien and Tibshirani [12]. Kim et al. [45] state that good prototypes are not enough to explain a model properly. By using the maximum mean discrepancy, a distance measure for distributions, they not only find appropriate prototypes, but also examples that are not well fitted by the model, so called *criticism samples*. Similarly, Duivesteijn and Thaele [25] are interested in understanding where a learned model is *not* working well. Their proposed model evaluator searches for subsets in the data that interact in an unusual way, i.e. differ from the ground truth.

A more mathematical surrogate is presented by Baehrens et al. [8]. They create explanation vectors, which are based on the derivative of the conditional prediction probabilities. Such vectors can be visualised in a surface plot, allowing to interpret which regions on the surface have the highest prediction probability.

While a surrogate model may help understanding the general reasoning of the underlying black-box model, it has several major drawbacks. For one, it often only correctly predicts the majority of instances, but neglects special cases. For another, it often simply can not capture the structure of the model space, thus becoming misleading. Imagine a simple decision tree. The decision function created by this tree can only contain decision boundaries parallel to an axis. Hence, if the model space of the underlying black-box model contains diagonal decision boundaries, the approximation tree will either classify instances incorrectly, or is overfitted. The domain expert may infer wrong conclusions from this approximation. An exception of this argumentation is the usage of local surrogates like LIME. They do not suffer from global approximation, as they only claim to be locally faithful.

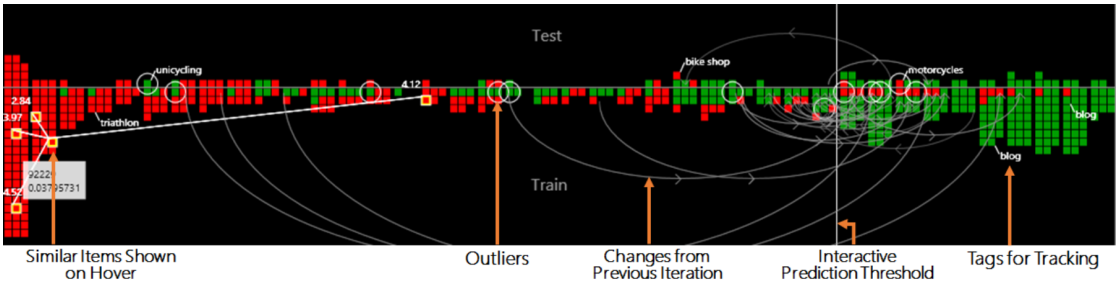


Figure 4.5: Visualisation of classified instances with ModelTracker [4].

4.4 Interactive Visual Analysis

Most of the approaches discussed so far provide concepts or algorithms for automatically computing a measure about how important a certain feature is, which feature combinations are meaningful, or suggest interpretable surrogate representations. Only few publications provide interactive visual analysis interfaces, which allow users to modify feature values or focus on specific regions and immediately see the results of their interactions in the framework.

Several interfaces can be found that support the user in analysing the general performance of a classifier. Exemplary, Amershi et al. [4] present *ModelTracker*, an interactive visualisation for the analysis of learned binary classification models. To gain an overall understanding of the model performance, several well-known evaluation metrics such as the ROC curve and confusion matrix are visualised. For a more detailed inspection, they show the classified instances as coloured boxes in a panel, ordered by their assigned prediction probability (see Figure 4.5). Using the colour coding and hovering over instances, the user can analyse how instances are classified, which instances are misclassified and find similar instances. Interfaces like ModelTracker can improve the general understanding in a classification model, but do not specifically support the user in understanding the reasoning for a prediction.

A more specialised interface is developed by Kulesza et al. [55]. Their interactive machine learning system explains to the user the reasoning behind a prediction and allows to manually correct a false prediction. The framework is restricted to text classification by multinomial naive bayes. It explains a classification by showing the most influential words and a-priori probabilities using bar and pie charts. An example is shown in Figure 4.6. Similar work can also be found for other model types, for example deep neural networks [41]. Such interfaces provide great visual support for the explanation of classifiers, but rely heavily on their restriction to certain models and are merely adaptable to other model types.

A model-agnostic visual analytics tool for model interpretation is presented by Krause et al. [50]: *Prospector*. It combines multiple new ideas. The partial

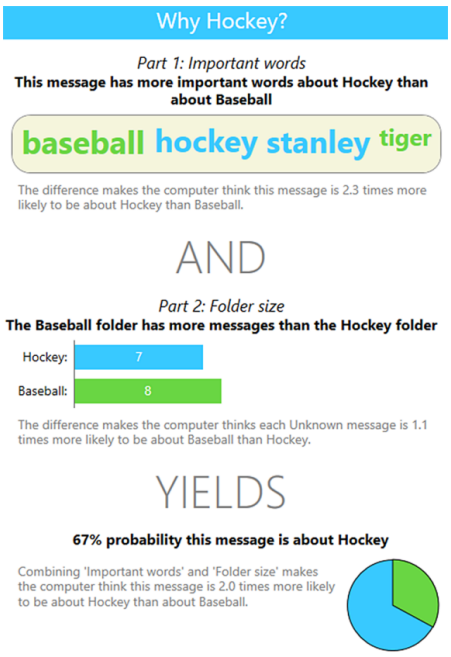
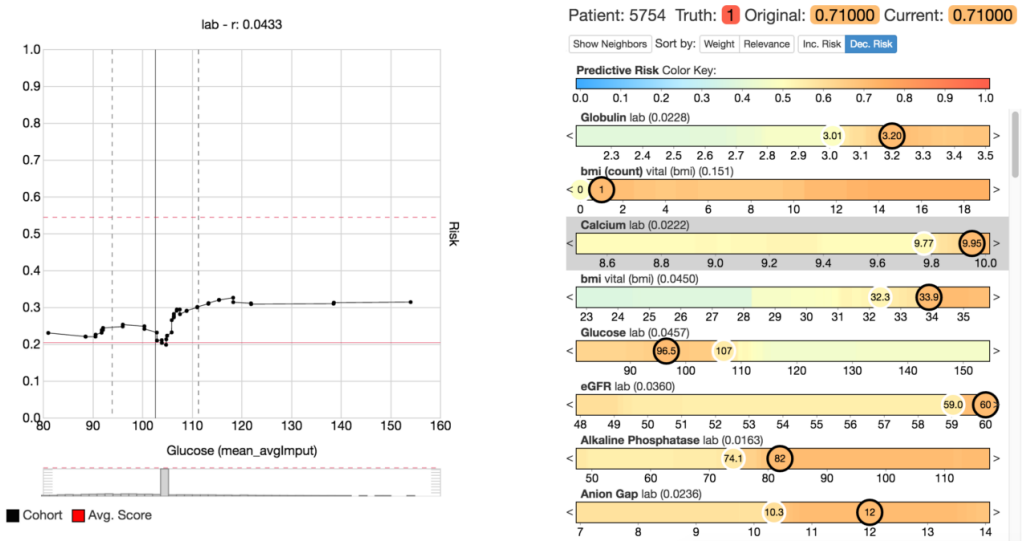


Figure 4.6: Explanation of classifying a message as *Hockey* [55].

dependence plot is enhanced by a smart sampling strategy and a histogram of the feature value distribution (see Figure 4.7(a)). It also supports multiple curves per plot, which enables it for comparison between different models. Using additional sliders called partial dependence bars, a user can change any feature value of a single instance and observe, how its prediction and PD changes (see Figure 4.7(b)). A new local feature importance metric is introduced that measures, where a small change in the feature value causes a significant change in the prediction. Prospector is an advanced framework for model explanation supporting both global and local analysis. However, Prospector's ability for model comparison is somewhat limited to multiple curves in the PDP. In addition, for data sets with many attributes the framework becomes confusing due to the number of partial dependence bars. Furthermore, Prospector only allows the analysis of binary classifiers.

In another work, Krause et al. [51] introduce a workflow for visual diagnostics of binary classifiers. Their framework is divided into three areas: A statistical summary view, a global explanation view, and an instance-level inspector. The summary view provides general information about the classifier's performance: A confusion matrix, ROC curve and a histogram of the prediction score distribution. In the global explanation view, explanations generated by the procedure of Tamagnini et al. [81] are displayed along with some statistics such as a confusion matrix. This panel is shown in Figure 4.8. The most granular inspection of the model is



(a) Enhanced PDP **(b)** PD bars

Figure 4.7: Prospector. (a): The PDP uses a sampling strategy based on the data distribution, as shown in the histogram beneath. (b): Partial dependence bars allow to inspect the effect of modifying a variable [50].

available in the third panel, the instance-level inspector. Here, an unordered matrix of all instances and features is shown, allowing to find patterns in items or find information about misclassified instances. The framework only supports binary input data and classification.

A visual interface specifically designed for the comparison of classification models with respect to model interpretation is proposed by Kahng et al. [40]. They argue that comparing two models just by their overall predictive power, i.e. accuracy or other metrics, is often too coarse, but only inspecting individual instances is too time-consuming and not scalable for big data sets. They suggest a compromise by comparing models on a subset level: The data is split into different subgroups, for example by age group or sex. The quality of the models is then compared by visualising how they perform in each subset. The user can specify multiple and custom subsets, or change the evaluation metrics (see Figure 4.9). The presented approach is not limited to binary data, but can handle both numerical and categorical data. It also theoretically supports multi-class prediction, although this has not been tested by the authors. While the main contribution of it is to discover well-predictable data subsets, it does not necessarily explain any prediction; Instead, it could be also categorised as a method for subspace discovery.

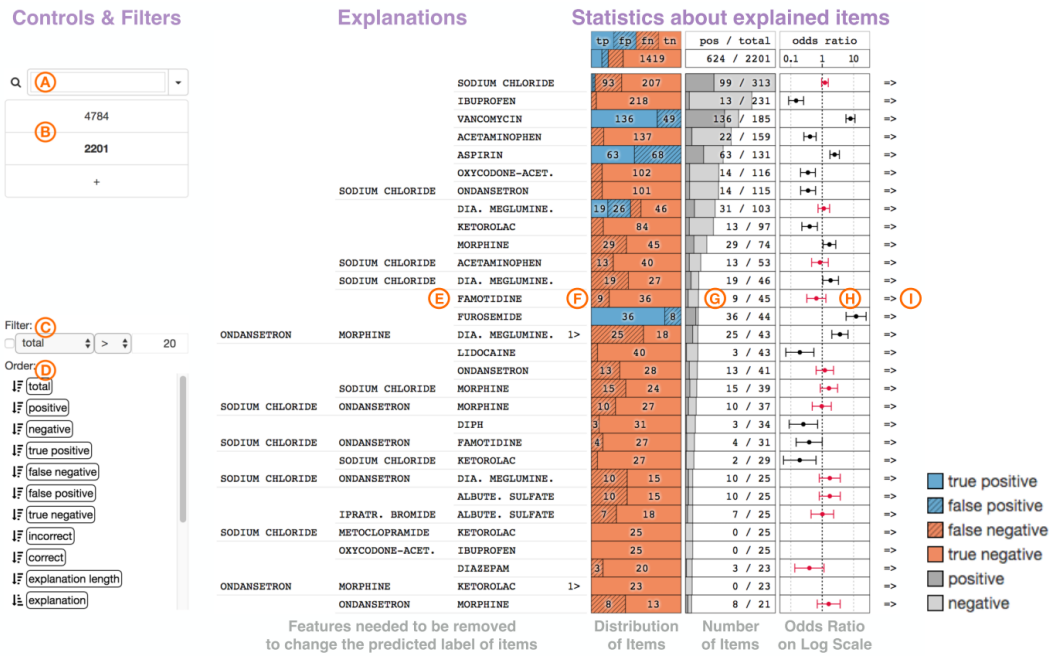


Figure 4.8: Visualisation of binary explanations in Krause et al. [51].

Figure 4.9: Comparison of classification models via subsets [40].

4.5 Summary

This chapter provided an overview of existing literature in the area of model interpretation, focusing on model-agnostic interpretation and the explanation of tabular data, as required by Requirements 1 and 3. Related work is concerned with assessing the importance of a feature, finding meaningful feature combinations, and approximating a black-box model with an interpretable surrogate model. While some previous work is restricted to binary data or binary classification, other approaches appear to be well-applicable for the interpretation of classifiers trained on epidemiological data. Visual interfaces for interpretation are rare; They often only provide a general understanding of the classification performance, or are limited to certain model types. No work has been found that specifically addressed the challenge of interpreting data with longitudinal information (Requirement 2).

Summarising these findings, the development of a novel visual interface to specifically interpret classifiers trained on cohort study data – in order to satisfy the declared requirements – seems to be justified.

5. Visual Interface for Model Interpretation

5.1 Overview

The previous chapter gave an excursus to the research field of model interpretation and visual interfaces for model interpretation, and discussed that existing visual approaches do not satisfy the requirements declared in Chapter 3. This work presents a new visual framework that targets previous shortcomings. Its key features are:

- Analysis of binary, categorical and numerical features
- Support of longitudinal features
- Classification into binary and multi-class target attributes
- Model-agnostic interpretation at global and local level
- Comparison of two independently learned models

An exemplary overview of the proposed framework is shown in Figure 5.1. As can be seen, the interface can be roughly divided into four parts, each pursuing a different objective:

1. The data panel at the top left allows to select two data sets to be analysed independently from each other, as well as a classification model to be learned

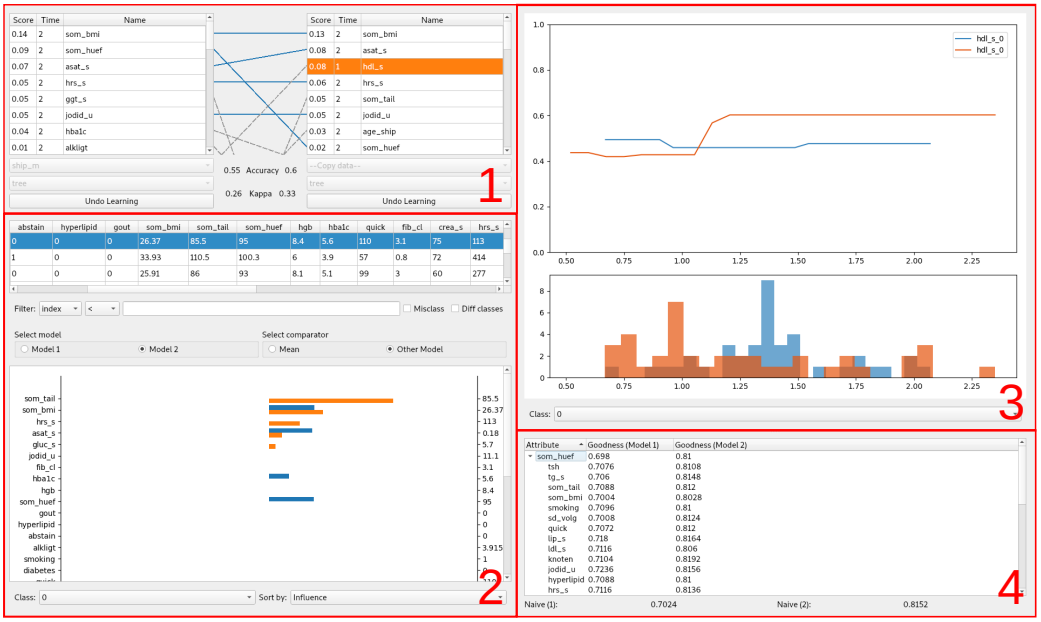


Figure 5.1: Overview of the proposed visual interface. The shown data set is described in Section 6.2.

on each data set. It supports a detailed selection of the attributes to be considered in the learning process by giving the user the option to remove attributes from the data or to replace them by related attributes. Once a classifier is learned, this panel shows two general metrics about the classifier's performance as well as global feature importance scores, and ranks the attributes based on their scores. If both models are learned, the panel compares the score rankings with a bipartite graph.

2. The individual importance panel allows for a detailed analysis of individual instances. It displays all instances in a table and provides a simple filtering mechanism, including functionality to show only incorrectly classified instances or instances that were classified differently in the two models. A horizontal bar chart displays the selected individual's feature importances per class and allows to compare them with the mean importances as well as the feature importances of the other model.

3. The partial dependence panel shows an enhanced partial dependence plot and a histogram of the feature distribution. It supports the visualisation of numerical and categorical features, and is suited to multi-class classification scenarios.

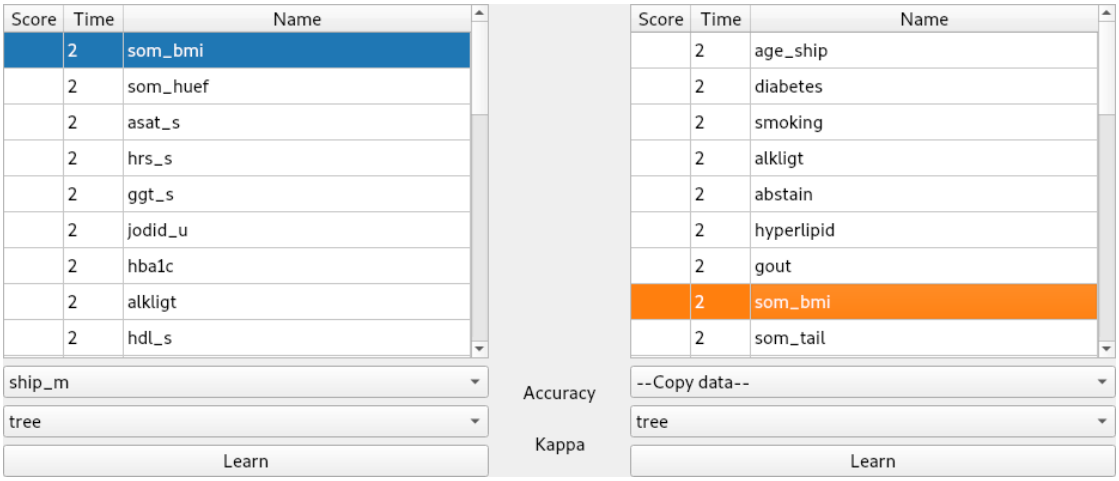


Figure 5.2: Data selection panel. Left and right area each allow to select a data set. Its attributes are shown in a list, allowing for modification. After modifying the data, the user can select and train a classification model.

4. The feature grouping panel contains a novel approach to detect meaningful feature combinations. A tree view presents the intermediate results of the analysis and ranks feature groups based on their "combination goodness". The user can choose to increase a group by an additional feature or to analyse other possible groups.

The following sections cover the functionality of the four panels in more detail.

5.2 Data Selection

Amershi et al. [3] state that users like to experiment with the input of a model and compare the output of multiple models. This not only allows to find the best model, but also to investigate the change of small modifications in the data. Following this argumentation, the proposed visual interface provides two widgets, in order to learn *two* classifiers on *two* (potentially different) data sets (Figure 5.2, left and right area). If not specified otherwise, the following descriptions apply to both widgets.

At the start of an analysis the user selects a data set. The visual interface supports any data in tabular form, i.e. binary, categorical and numerical features[1]. Once a data set is selected using the upper combo box, its attributes are displayed in the

[1]Binary features are a special case of categorical features. In the remainder of this work, descriptions of categorical data include binary features.

table. The table contains three columns: (1) The *Score* column displays the global feature importance score of each feature, once the classifier is learned; (2) In cohort studies, examinations of certain features are potentially repeated in follow-ups. If such information is available, the *Time* column displays the examination that will be used to train the classifier. By default, the latest examination is chosen. For features that do not have time-related multiple values, this column is redundant; (3) The *Name* of the feature.

Using the list view, the user can inspect the existing features and manually modify them. The available operations for modification are:

Remove a Feature If the user removes a feature from the data set, the feature is not given to the classifier and therefore not considered in the learning process. If the feature has values of multiple examinations, none of them are considered. This operation may be useful if the user has prior knowledge about the attribute, or wants to analyse if the classifier can compensate the missing information using other features. This becomes especially relevant, if the removed feature is expensive to acquire, for example only via f-MRI.

Replace a Feature If a feature has multiple values from repeated examinations, the user can choose, which of the values he wants to give to the classifier. In the current implementation, such related values need to be of longitudinal nature; The underlying idea however also applies to features which share another kind of relation with each other. To clarify the usefulness of this operation, consider a feature which is expensive to acquire. If removing the feature does harm the classifier's quality, i.e. can not be compensated by other features, it may be still sufficient to use the result of a previous examination, thus saving the efforts and expenses of a fresh acquisition.

To simplify the process of comparing two modified data sets, the data selector of the right widget contains an additional option to copy the current attribute information in the left widget and continue with further modifications. Throughout the framework, a consistent colour coding scheme is used, which allows to ascribe emphasised rows, lines and bars the respective classifier. The model specified in the left widget uses blue colours. The model specified in the right widget uses the complementary colour orange.

In addition to comparing the same data set with different selected features, the framework also has limited support to compare different data sets. This ability is intended to compare subsets of data sets with mostly congruent features, for example male and female members in a cohort. Naturally, not all parts of this interface support this type of comparison: Analysing the differences between the

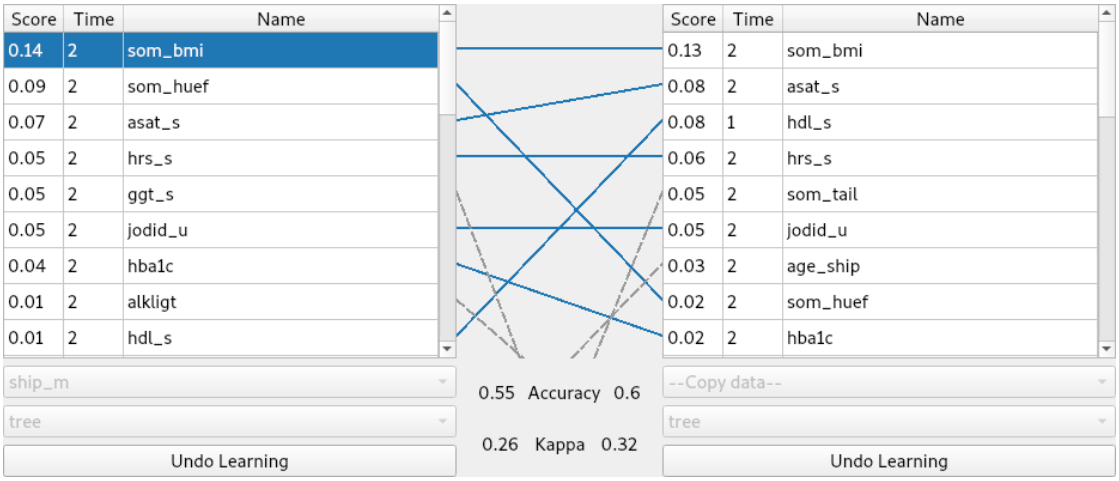


Figure 5.3: Feature importance ranking and comparison of two rankings. The features of each widget are ranked in descending order by their importance score. In between the lists are general model quality metrics displayed. A bipartite graph connects attributes between the ranked feature lists.

models on an individual level is only possible if an individual is present in both models.

As required by Requirement 3, the framework is model-agnostic and theoretically accepts any type of classification model. However, several techniques used in the analysis of feature importances are based on prediction *probabilities*, i.e. the probability of an instance belonging to a certain class. As a consequence, the objective of model-agnosticism has been softened to support any *probabilistic* classification model, or models which can simulate prediction probabilities[2].

5.3 Model Quality and Feature Importance

Model Quality

Once the classification model is learned, two measures are displayed in the panel, which allow for a first impression of the general model quality (see Figure 5.3): The accuracy as the most popular quality metric as well as Cohen's Kappa score as an imbalance-aware metric (see Section 2.2.1).

Feature Importance

In addition, the individual and global feature importance scores are computed. To allow a comparison, the importance scores for global and local importance should

[2]For example, the non-probabilistic classifier *k-Nearest-Neighbour* can return prediction probabilities by computing the ratio of each class in an instance's neighbourhood.

be computed by the same method. From the literature discussed in Chapter 4, two techniques have been identified as qualified for this requirement: LIME [72] and the approach by Robnik-Šikonja and Kononenko [74]. LIME learns simple, locally faithful models. To get a global understanding of the model's reasoning, a set of representative instances may be selected. Preliminary tests showed that the representative selection for LIME is not computationally efficient enough for the analysis of complex instance spaces, such as in cohort studies.

The algorithm by Robnik-Šikonja and Kononenko [74] computes individual importance scores by computing the prediction difference with and without a feature. They approximate a global importance by taking the average score over all instances, and compare local and global score in a bar chart. Their application however is limited to binary classification, which simplifies the averaging process, as only one score exists per instance and feature. In the multi-class scenario considered in this thesis, the local feature importance is characterised by one score per class. Therefore, an additional mean aggregation over all classes is required. Here, the absolute value of a score is used, as the class average would be mutually eliminated otherwise. The new formula for computing the global approximate of attribute $\mathcal{A}_i$ is:

$$\text{globalPredDiff}_i(x) = \frac{1}{|X|} \sum_{x \in X} \frac{1}{|C|} \sum_{c \in C} |\text{predDiff}_{i,c}(x)| \tag{5.1}$$

Robnik-Šikonja and Kononenko [74] propose three different formulas for the prediction differences. A preliminary evaluation of this work did not reveal any notable qualitative differences between the formulas, which is why the direct probability difference is used in the remainder of this work (see Equation 4.4). Here, the maximum importance depends on the original prediction probability, but can not exceed a value of 1. In follow-up studies, the importance score computation has been modified to determine the marginal importance, i.e. removing influences of conditional dependences between features. This extension requires iterating over the power set of feature combinations, making it infeasible from a computational complexity point of view.

Comparing Global Feature Importance Scores

Are both classification models learned and the feature importance scores computed, a bipartite graph visualises the differences in the importance ranking (see Figure 5.3). Blue lines connect a feature to its counterpart in the other ranking; Features which are currently not visible in the list are connected via grey, dashed lines. It therefore provides an intuitive way to get an early overview of the differences between the models.

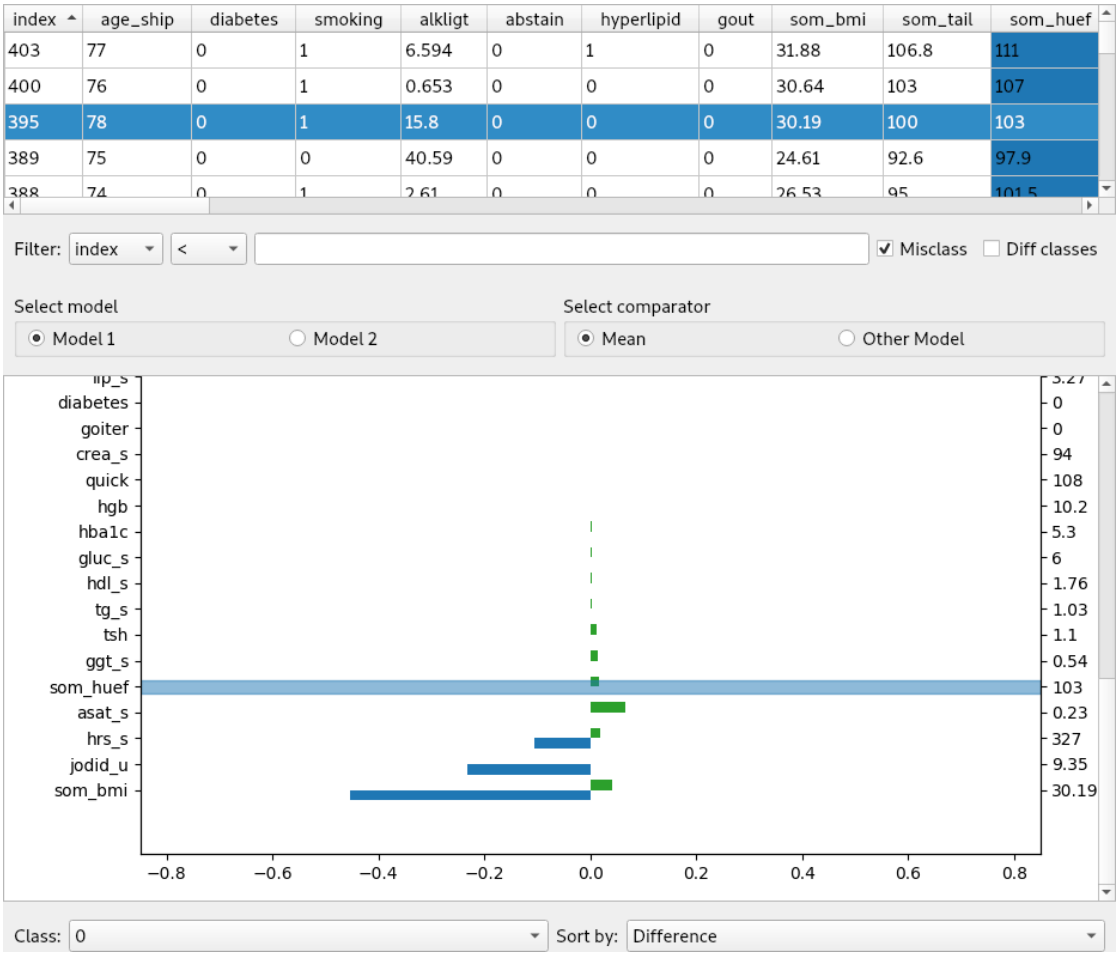


Figure 5.4: Individual analysis panel. Top: A table displays all individuals with their attribute values. Bottom: An enhanced explainVis plot shows the feature contributions for the prediction of a single individual.

The bipartite graph is a simple form of parallel coordinates, which have been already used in the literature to compare time-related ordered data [84]. Other work specialised on comparing ordered data is space-consuming and requires a training period to get accustomed with (for example Behrisch et al. [11]), and has therefore been rejected in favour of the bipartite graph.

5.4 Individual Analysis

The importance score rankings and their comparison allow to understand, which features contribute to the overall classifier reasoning. As already stated, often a local inspection is equally relevant to the expert. This is provided in a second panel.

It consists of two main parts: A table displaying all individuals and a visualisation showing the individuals' feature importance scores.

The table at the top of the panel shows each individual in the test set along with its feature values, its true label and its predicted label. A simple filtering mechanism allows the user to view only instances above, below or in the range of self-defined feature values. Additionally, the user can choose to only display incorrectly classified instances in the table, and instances which are classified differently between the two learned classification models.

Beneath the table a horizontal bar chart is displayed. Its design is based on explainVis [74], but has been extended in several ways:

Comparison to the Average and Another Classifier

The original visualisation plots two bars for each feature: The individual's feature importance and the average importance. This option is provided in the visualisation of this work, too. In addition, the user can replace the average importance with a bar displaying the feature importance in the second classifier. This requires that both classifiers are learned on the same data set. The comparison enables the user to analyse, how the classifiers differ on an individual level.

Ordering the Individual Importance Chart

Three ways of ordering the features in the data are available to the user:

Alphabetically This is the original sorting order for the features in the chart. It allows the user to quickly locate a feature in the graph based on its name.

Feature Importance This allows to directly overview, which features contribute towards a prediction, which features are not affected or even counteract a prediction. Additionally, in high-dimensional data sets such as epidemiological cohort studies, most features are often not used by the classifier at all. Here, the remaining important features are spread over the visualisation, making it hard to directly compare them. Ordering features by their importance prevents this, as it automatically clusters relevant features.

Difference As explained in the previous paragraph, the user can compare an individual's feature importance to the average importance or the other classifier. By ordering the data by the difference to the average score, features can be detected that were interpreted in another way for this particular instance than for others. This may indicate for an outlier or subpopulation.

Multi-Class Classification

Due to the limitation of the original algorithm to binary classification, the visualisation is not suited for multi-class settings, as it only shows the prediction difference with respect to one class. As a consequence, a combo box has been added to select the class, of which the prediction difference is displayed. While this solution does not display all given information at once, it may be seen as a step towards visual inspection for multi-class settings.

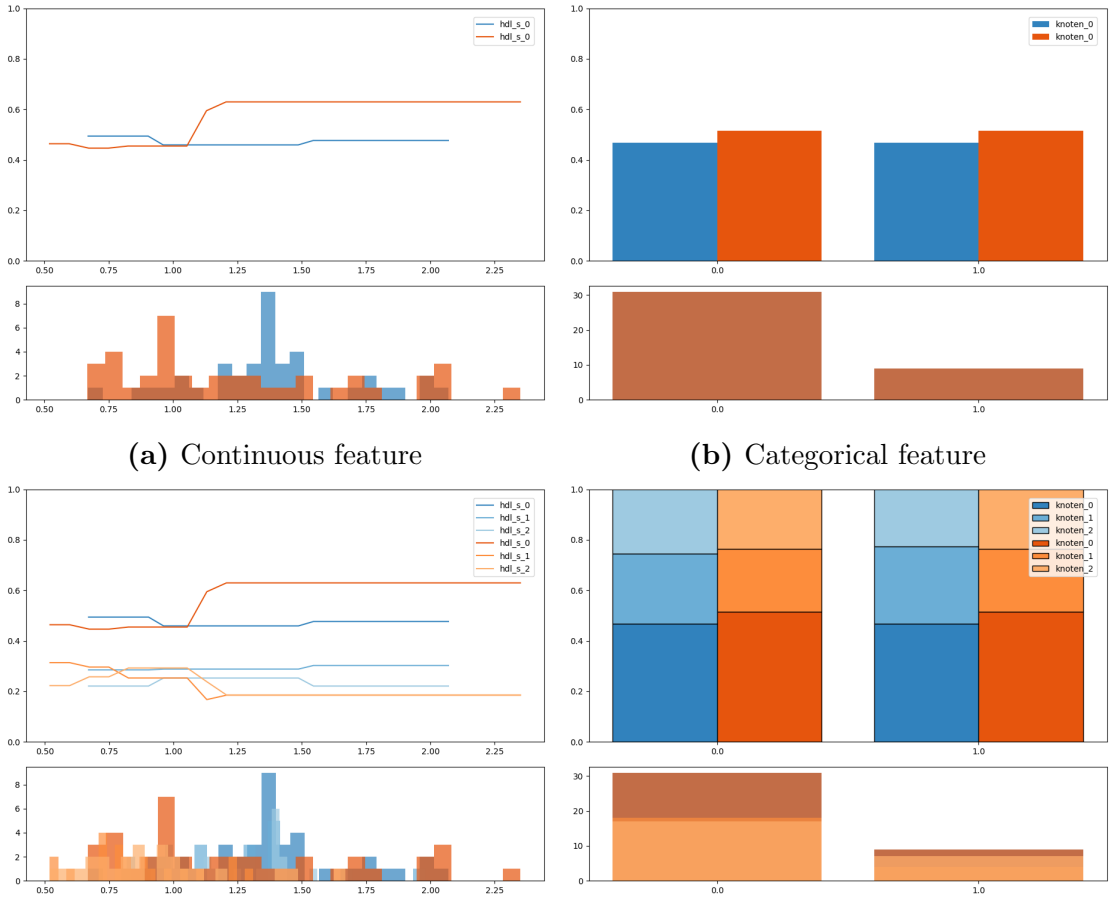
Highlighting a Feature

By selecting a feature in the global attribute list (see Figure 5.3), it is highlighted in the individual importances visualisation, allowing to quickly find the feature. This is useful in many-variate data sets.

5.5 Partial Dependence

The panels described above allow the selection and modification of data, as well as the inspection of global and individual feature importance scores. An importance score itself however does not give an insight into the *relation* between the feature's value and the prediction probability. In Chapter 4, the partial dependence plot and related charts have been introduced, which visualise such relationship. This framework uses the basic PDP, as presented by Friedman [29] instead of more advanced techniques. The partial dependence plot is standard in the literature and has the advantage of a simple, intuitive computation. Thus, it does not require intensive training to the expert. Nonetheless, it has been slightly modified to fit the requirements of this application:

- The PDP supports categorical data by switching to a bar chart whenever a categorical feature is inspected (see Figures 5.5(a) and 5.5(b)).
- Similar to the individual feature importance visualisation, the PDP comes with a combo box to select, of which class the prediction probabilities are to be shown. The user may also see the partial dependencies for all classes simultaneously. In this case, categorical features are displayed in a stacked bar chart (see Figures 5.5(c) and 5.5(d)).
- A histogram beneath the PDP shows the data distribution, which gives an understanding of which values are common, or whether there are outlying subgroups. This concept is borrowed from Krause et al. [50], but modified to see only the data distribution of the inspected class (or all classes, if selected).
- Different features can be directly compared in the same chart. This is motivated by analysing features which share the same unit and range, for example hormone levels or age-related conditions.



(a) Continuous feature **(b)** Categorical feature

(c) Continuous feature, all classes **(d)** Categorical feature, all classes

Figure 5.5: Available variations of the partial dependence plot and distribution histogram.

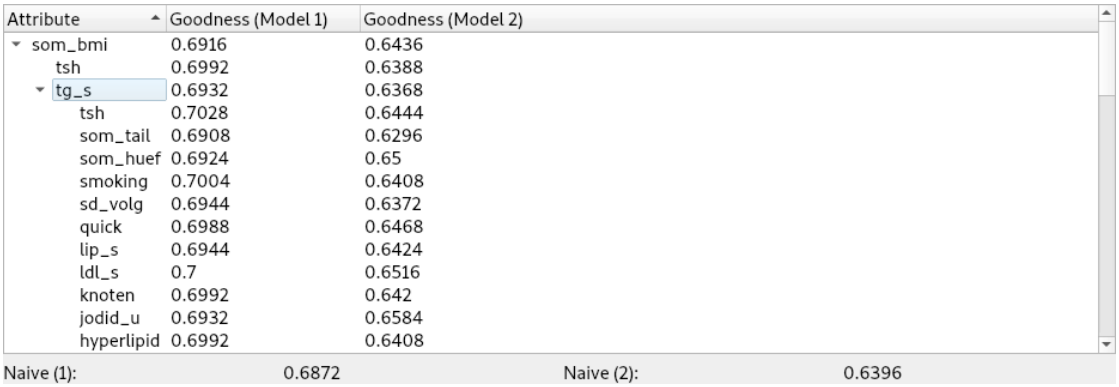


Figure 5.6: SilverEye visualisation. The goodness at a given feature represents the model's fidelity for the group containing the feature and all of its parents.

5.6 Feature Grouping

As argued by Henelius et al. [35], the discovery of meaningful feature combinations is a desirable ability in interpreting a classification model. Their work finds the optimal grouping of the features in a data set, but is computationally inefficient in high-dimensional data sets. To integrate their idea into the framework of this work, an interactive and iterative approach has been developed. It does not aim to find the overall best grouping, but tries to find the best group for a given feature. In reference to the name of the original algorithm, the modified version will be named *SilverEye*. It works in the following way:

The original algorithm groups all features together and removes one feature at a time, until the goodness criterion approximates the naive goodness, where only singletons exist (i.e. only groups of size 1). In contrast, SilverEye starts at the naive goodness. For a selected feature $\mathcal{A}_i$, it computes the goodness values for all possible groups containing $\mathcal{A}_i$ and a second feature. All other features remain as singletons. The resulting values are displayed in a tree structure and can be sorted alphabetically or by goodness (see Figure 5.6). This way the user can inspect which features performed well together with $\mathcal{A}_i$. To continue with the analysis, the user can select a feature $\mathcal{A}_j$ in the new tree level, which will cause the algorithm to compute all goodness values for the group $\{\mathcal{A}_i, \mathcal{A}_j\}$ and a third feature. Alternatively, he can start with a new feature. SilverEye has no termination criterion, as it is user-driven. Instead, the user may decide himself, if he wants to look further for more group members or not. Ending an analysis may be reasonable if the group is satisfied, i.e. the goodness value does not increase significantly any more by adding a feature. As a hint, the naive goodness value is displayed below the tree. It should be noted, that SilverEye is based on randomisation. Thus, goodness values can vary for identical groupings, especially if the classifier does not perform

well or the test set is small. This also infers that values lower than the naive goodness may be computed. SilverEye comes also with the advantage of integrating expert knowledge: The user may reject the best performing attribute in favour of another attribute, if the best feature is a known redundancy or expensive to acquire. SilverEye also allows the free choice between exploration and exploitation, as the user can investigate other features at any time.

The goodness function used in SilverEye is the fidelity, a measure introduced by Henelius et al. [35]. In GoldenEye++ [36], an alternative goodness metric has been developed that uses the correlation between probability vectors. However, this is restricted to binary classification. During the work on this thesis, an extension of GoldenEye++ for the multi-class scenario has been tested, which compared multi-dimensional probability distributions instead of single probability vectors based on distance correlation [80]. Unfortunately, it resulted in a high computational complexity and is thus not a reasonable choice for the framework.

5.7 Technical Details

The proposed framework has been developed in the script programming language Python[3]. Data sets are loaded and managed via pandas[4], which supports database-like and vectorised operations. Additional scientific computations, for example in the computation of the feature importance scores, are realised via numpy[5]. The individual importance bar chart, the partial dependence plot and its histogram are created via matplotlib[6]. The available classification models are implemented with scikit-learn[7]. An exception is gradient boosting, for which the XGBoost[8] implementation is used. The graphical interface is built with PyQt5[9], a Python port for the application framework Qt5.

The source code is freely available at https://github.com/tsabsch/vmi. Functionality for SilverEye is provided in the `goldeneye` package, which can be found at https://github.com/tsabsch/goldeneye.

[3] https://www.python.org/, Version 3.6
[4] https://pandas.pydata.org/, Version 0.22
[5] https://www.numpy.org/, Version 1.14.2
[6] https://matplotlib.org/, Version 2.2.2
[7] http://scikit-learn.org/, Version 0.19.1
[8] https://github.com/dmlc/xgboost, Version 0.71
[9] https://pypi.python.org/pypi/PyQt5, Version 5.10.1

6. Evaluation

This chapter describes the evaluation performed on the proposed visual interface. First, it briefly introduces the methodology of evaluating visual analytics interfaces and provides details to the used classification design. Then, the evaluation is performed. It consists of two case studies, one being a real-world epidemiological cohort study, and the other being a synthetic data set. Finally, the results of the evaluation are discussed.

6.1 Prerequisites

6.1.1 Methodology

The following description on evaluation methodology is based on the explanations given by Keim et al. [43].

Proper evaluation in visual analytics is challenging, due to the broad range of aspects that can be evaluated. An evaluation should concern the "assessment of the quality of artefacts" [43], where artefacts include for example analytics techniques, models or theories. The key aspects for the quality of an artefact are *effectiveness*, *efficiency* and *user satisfaction*. In other words: Do the artefacts fulfil their aims with reasonable resources and meet the expectations of the end user? Evaluating these aspects is not always trivial: While an evaluation of the effectiveness can be performed straight forward (*Does the technique produce the desired results?*), other questions are hard to quantify, such as a comparison to existing tools.

Keim et al. [43] introduce several methods for the evaluation of interactive techniques: *Quantitative* and *qualitative methods* primarily analyse the effectiveness

and efficiency of interfaces using controlled test environments. *Usability studies* on the other hand analyse the ease of using the interface. *Informal evaluations* discuss with users about their experiences with the tools, allowing for feedback.

The novel interface presented in Chapter 5 will be informally evaluated in terms of effectiveness. Here, the question to be answered is: How useful is the interface for the generation and validation of epidemiological hypotheses?

To answer this, a real-world epidemiological cohort study is analysed. This simulates the application of the interface in a productive environment. The steps in the analysis will be described, and the generated findings about the cohort study presented. A comprehensive analysis of the real-world data in order to test the quality of the interface for exploration (e.g. generation of hypotheses) is not feasible, due to the high-dimensionality of the data. Instead, a second case study on an artificial data set is performed. This complements an explorative analysis.

A usability study and informal evaluation with an epidemiological expert are *not* part of this evaluation. This is due to time constraints, but should be addressed in future work.

6.1.2 Classification

Classification Models

The presented interface is model-agnostic in theory. As discussed however in the previous chapter, several techniques require probabilistic classifiers in order to work. In this evaluation, five classifiers have been implemented and are available for use: A decision tree, random forest, gradient boosting, logistic regression and a neural network. A description of these models is given in Section 2.2.2.

All of these classification models require certain hyperparameters, which have been empirically determined in preliminary tests. As an explanation and review of all parameters would exceed the scope of this evaluation, they are only briefly presented in the following for sake of reproducibility:

Decision Tree The algorithm used for the tree induction is CART [16]. To assess the information gain at each node, the *gini* split criterion is used. For regularisation purposes, the tree is restricted to have a maximum depth of five and at least four samples per leaf. To obtain class probabilities instead of single predictions, the fraction of instances with the same class within a leaf is used.

Random Forest Critical hyperparameters for the performance of a random forest are the number of decision trees and the fraction of features randomly selected

Table 6.1: Hyperparameters used in the neural network

Parameter	Value
Activation function	Logistic function
L2 penalty (α)	0.1
Learning rate	Adaptive
Number of hidden layers	2
Number of hidden units	20, 4
Solver	Adam [47]
Number of epochs	2000

in each data subset. Although a forest is not sensitive to overfitting with an increasing number of trees, it affects the learning time and should be chosen carefully. In this evaluation, 100 trees and a maximum fraction of features per subset of 50% are used.

Gradient Boosting The gradient boosting classifier is built upon 1000 decision trees. The loss function to be minimised is the probabilistic softmax criterion [30]. The used implementation XGBoost only supports binary categorical attributes, which is why categorical attributes are one-hot-encoded.

Logistic Regression The optimisation of the logistic coefficients is done with SAGA [23] and Lasso regularisation [83]. The inverse penalty factor is set to $C = 1$. Equivalently to gradient boosting, categorical features are one-hot-encoded.

Neural Network Neural networks require various hyperparameters, such as the number of hidden layers, number of units per layer or learning rate. A list of the chosen parameters is given in Table 6.1. For parameters not listed in the table, the default parameters in scikit-learn are used.

Train / Test Split

Once the user has specified his desired data set and type of classification, a classifier is trained. In before, the data set is randomly split into a training and a test set, where 80% of the data is used for training and 20% of the data for testing. A third validation set, which can often be found in machine learning applications for hyperparameter optimisation, is omitted in this work, as parameter optimisation is not within the scope of this work. Other common techniques for assessing the general quality of a classifier, such as bootstrapping or cross-validation, are

not applicable in this framework, as they average the quality of *multiple* models. The proposed interface however analyses a *single* classification model. To ensure comparability between learned models, the splitting function uses a constant random seed. This also contributes to reproducible results. Furthermore, the random split is stratified with respect to the class, i.e. the class proportions are preserved in both training and test data.

6.2 Case Study: Study of Health in Pomerania

The first case study interprets a classification model which is learned on a real-world epidemiological cohort study. The target of the classification is to detect, whether a study participant has *hepatic steatosis* (short: HepStea, and common: Fatty liver), which is an increased fat percentage in the liver. A medical introduction into the risk factors of hepatic steatosis is for example given by Bedogni et al. [10].

Before the proposed interface is evaluated on the data, a brief overview of the cohort studies and previous investigations is given, as well as a description of the preprocessing steps.

6.2.1 Study of Health in Pomerania

The study of health in pomerania (SHIP) is a population-based cohort study. Designed and managed by the Institute for Community Medicine, University of Greifswald, Germany, it investigates the health conditions of the northeast German region Pomerania [87]. SHIP has two main objectives: (1) To assess the prevalence and incidence of risk factors, subclinical disorders and clinical diseases; (2) To investigate complex associations among risk factors, subclinical disorders and clinical diseases.

An outstanding property of SHIP is, that it does not address a specific disease, but rather generally describes health-related conditions. As a result, the attributes contained in the study are highly heterogeneous: Acquired features contain, but are not limited to personal interviews, laboratory tests, ultrasound, dental examinations, f-MRI and sleep monitoring. Follow-up examinations are performed every five years. At the time of this writing, SHIP contains the baseline examination SHIP-0, which was assessed between 1997 and 2001 with 4308 eligible subjects, and two follow-up examinations: SHIP-1 (2002-2006, 3300 subjects) and SHIP-2 (2008-2012, 2333 subjects). A second cohort called SHIP-TREND has been created from 2008-2011 with an initial population of 4420 [87].

Previous Studies

The Study of Health in Pomerania has been extensively analysed, for example with respect to the prevalence of health conditions like overweight, gall stones, arterial hypertension or hepatic steatosis [87]. At the Otto-von-Guericke University Magdeburg, Germany, SHIP has been analysed with data-driven approaches and visual analytics to support epidemiologists in classifying participants and finding subpopulations. While a comprehensive overview of previous studies would extend the scope of this section, few approaches may be mentioned which particularly address the longitudinal nature of the cohort study.

Hielscher et al. [38] propose a mining workflow for longitudinal epidemiological data. The key of their approach is the generation of *sequence features* for each participant. First, they create for each instance and feature a sequence containing the feature's values of all examinations, along with the associated class. Then, the sequences are clustered via DBSCAN. The cluster ID of each sequence is added to each individual as a new sequence feature. Niemann et al. [68] construct so-called *evolution features* to improve a classification task, as they assume that similar participants evolve similarly. The evolution features are generated by grouping the participants at each moment on similarity. The clusters and participants are traced on the time axis, thus capturing their *evolution*. Then, the cluster membership information and several cluster properties are added to the individuals as their evolution features. Mayer [64] presents a visual analytics interface to find and validate subpopulations. Using a classification rule miner developed by Niemann et al. [67], he compares the development of a subpopulation to the entire cohort's development using line, bar and box plots. This allows for further validation of the subpopulation's significance.

6.2.2 Data Preprocessing

The SHIP data used in this thesis consists of 886 participants, and 251 different features. In order to use it in the evaluation, several preparation steps have been necessary, which are described in the following.

Several attributes have been removed, which are uninformative for the given classification task:

- Attributes containing the same information, for example a participants' age.
- Date and location of an examination.
- Blood withdrawal times.
- Mortality information, such as life duration or main cause of death.

Table 6.2: Class distributions of $SHIP_M$ and $SHIP_W$

	$SHIP_M$		$SHIP_W$	
Class	Absolute	Relative	Absolute	Relative
0	198	46	321	70
1	110	26	65	14
2	118	28	74	16

– Attributes containing the status of hepatic steatosis. As the classification task for this evaluation is to predict HepStea, the data set may not directly address this condition.

Several participants are not examined on all attributes. To include them in the analysis, missing values are imputed, i.e. derived from the feature's distribution, which is approximated from the remaining participants. For this case study, it is sufficient to simply use the most frequent item in the feature distribution, called *mode*. There are however more advanced (but also more extensive) approaches available (see for example Alemzadeh et al. [2]).

One of the main considerations in this work is to test, whether using a previous examination of a value suffices to achieve reasonable classification results. As a consequence, only features are considered in this work, which have examination values for all recordings, i.e. SHIP-0, SHIP-1 and SHIP-2.

The target value is computed via the attribute `mrt_mean`, which contains the average liver fat percentage in a participant, acquired via an MRI in SHIP-2. Reddy and Rao [71] explain that hepatic steatosis is diagnosed if the lipid content in the liver exceeds 5-10% of its weight. Therefore, `mrt_mean` is discretised into three classes: Class 0 represents low fat percentage (`mrt_mean` $< 5\%$), class 1 represents medium fat percentage (`mrt_mean` between 5 and 10%) and class 2 represents high fat percentage (`mrt_mean` $> 10\%$).

Partitioning

Previous studies detected substantial differences in the class distributions between the sex [37], which is why the data set is split into two subcohorts containing the male (in the following denoted as $SHIP_M$) and female study participants (in the following denoted as $SHIP_W$), respectively. Table 6.2 shows the class distributions for both subcohorts. As one can see, male participants are more likely to have fatty liver than female participants.

In addition to different class distributions, the sex-specific partitions also differ from each other with regard to the available attributes. Female participants were additionally asked about previous pregnancies, menopause-related information and hormone replacement therapy.

Summary

Summarising the preprocessing steps, the final data set has been partitioned into two subcohorts: $SHIP_M$ contains 426 participants with 42 unique features (each consisting of three examination values), $SHIP_W$ contains 460 participants with 46 unique features (each consisting of three examination values). A complete list of all features is given in Table A.1.

6.2.3 Evaluation

This section contains an informal evaluation on the interface's usefulness for analyses on epidemiological data. An exemplary research question has been formulated, which is to be answered: *How do biomarkers in blood serum influence the prediction of fatty liver? Are recent laboratory tests necessary?*, given that such tests are more expensive than simple somatometric features or socio-demographic information.

General Classification Quality

As a first step, each classifier is trained on both subsets to determine, which model works best without further modifications. The attributes are not changed: All available attributes are used with their latest examination value (SHIP-2). While the interface is model-agnostic and therefore works on any of the available model types, the starting point for an insightful interpretation is a well-performing classifier. Table 6.3 presents the resulting quality metrics. As one can see, the female population is classifed better by all models, likely caused by the difference in class distribution. The male population achieves only poor accuracies. Apparently the data does not contain enough information for the classifier to properly distinguish the classes. The kappa values estimate the agreements between classifier and ground truth as being fair and moderate [56], which is a relatively weak result.

It should be pointed out that these classifiers are inferior to the approaches of Hielscher et al. [37, 38]. However, their algorithms are dedicated to achieve a high classification accuracy. Contrary to that, this study does not focus on achieving the highest-ever classification quality, but instead only uses the quality metrics to get a first general impression.

The best classifier, among both subpopulations, is gradient boosting. In the following, an exemplary interpretation of this classifier, trained on $SHIP_W$, is described in order to test the research question.

Table 6.3: Classification results on SHIP

	$SHIP_M$		$SHIP_W$	
Model	Accuracy	Kappa	Accuracy	Kappa
Decision Tree	0.55	0.26	0.62	0.17
Random Forest	0.59	0.33	0.75	0.37
Gradient Boosting	**0.62**	**0.37**	**0.78**	**0.44**
Logistic Regression	0.56	0.27	0.73	0.31
Neural Network	0.57	0.30	0.71	0.32

Score	Time	Name
0.08	2	som_tail
0.07	2	tg_s
0.05	2	ggt_s
0.04	2	age_ship
0.03	2	som_bmi
0.03	2	hgb
0.03	2	hrs_s
0.03	2	gluc_s

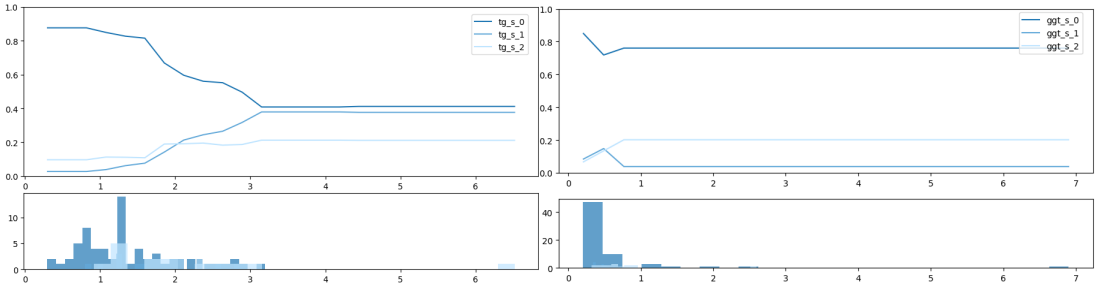
Figure 6.1: Global feature importance scores

Feature Importance

Figure 6.1 presents the ranked list of global feature importance scores, which allows to get a first insight into which features have contributed to the classifier. As one can see, all importance scores are relatively small. This indicates that the classifier does not heavily rely on the value of a single or few features, but uses a variety of contributing factors. The most important features to the classifier are somatometric features like waist circumference and body mass index, as well as age and biomarkers in blood serum. This is mostly in compliance to the findings of Bedogni et al. [10].

Partial Dependence

To gain a further understanding of *how* the features influence the classifier, the partial dependence can be used. With respect to the research question on the influence of blood serum biomarkers, the two most important blood biomarkers are analysed: Triglyceride (`tg_s`) and gamma-GT (`ggt_s`) level. The partial dependence plot for triglyceride (Figure 6.2(a)) shows that the probability of



(a) `tg_s` **(b)** `ggt_s`

Figure 6.2: Partial dependence plots of `tg_s` and `gg_t`. All classes are displayed.

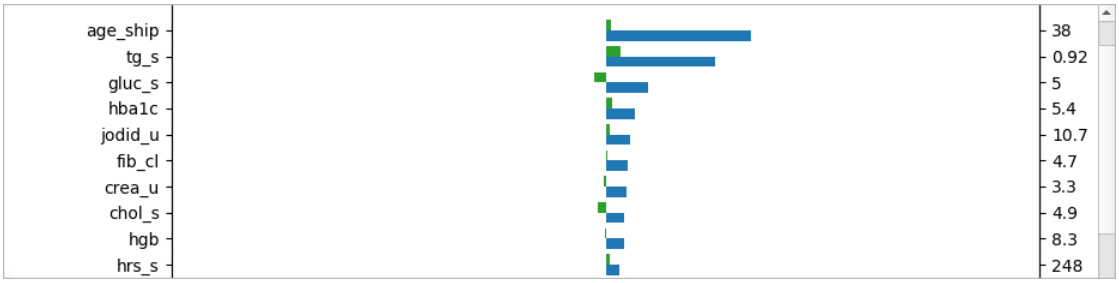
predicting class 1 increases with an increasing `tg_s` value, while the probability for class 0 decreases. Due to an outlier, parts of the PD curve are uninformative. This sensitivity towards outliers is even more apparent in the PDP of gamma-GT (Figure 6.2(b)). Here, an outlier removal before the analysis would achieve more insightful PD plots. It can be seen however, that the a-priori probability of not having fatty liver prevails (class 0).

Feature Combinations

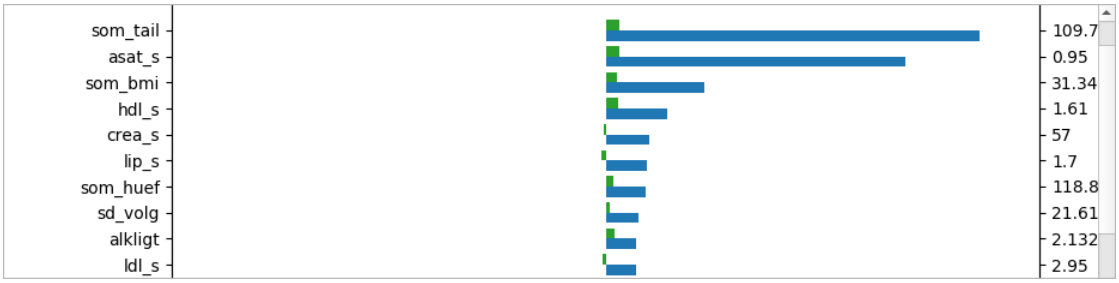
One of the aims of the proposed framework is to analyse, whether grouping features together affects the prediction performance. Based on the algorithm GoldenEye, a user-driven strategy called SilverEye has been developed in order to reduce the computational complexity. However, in this evaluation no results could be obtained in reasonable time. This shows that SilverEye is not applicable to data sets such as $SHIP_W$ with 46 attributes. A further analysis of its usefulness had therefore to be omitted.

Individual Analysis

A promising technique of the proposed interface is to analyse, how single participants have been classified. Here, especially the investigation of incorrectly classified instances suggests to be useful. Exemplary, the participant with ID 487 is selected. She does not have hepatic steatosis, but has been classified as belonging to class 2, i.e. having a high liver fat percentage. The explainVis plot reveals the reasoning behind this misclassification: The prediction of class 0 is mostly supported by her relatively young age and low triglyceride level (`age_ship`=38, `tg_s`=0.92). This is visualised in Figure 6.3(b). In comparison, predicting the participant as having a high liver fat percentage is mostly carried by a high waist circumference and aspartate enzyme level (`som_tail`=109.7, `asat_s`=0.95). This is visualised in Figure 6.3(a). The features indicating a high liver fat percentage are considered more important by the classifier, which likely caused the corresponding classification.



(a) Class 0



(b) Class 2

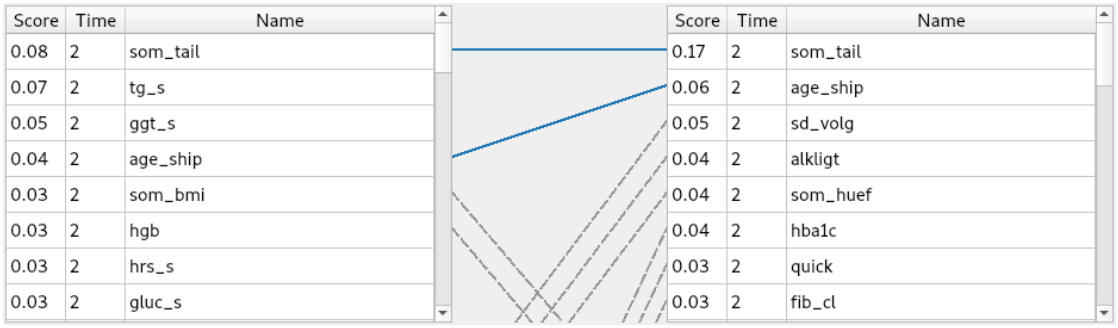
Figure 6.3: Feature contributions of participant 487 for classes 0 (a) and 2 (b).

Comparing Classification Models

As has been seen in the global importance ranking as well as in the individual analysis, biomarkers in the blood serum are relevant to a classifier. It could be however possible, that the classifier does not rely on these attributes, but can replace them if they are not available. Thus, a second gradient boosting classifier is trained, where all blood serum features are removed in before. It turns out that this model can not fully compensate the loss of the attributes, but still achieves an accuracy of 0.75 and a kappa score of 0.41. The bipartite graph connecting the importance score rankings shows, which features gained and lost importance (see Figure 6.4). The new classifier still uses a participant's waist circumference as its most importance feature. The new global importance score of `som_tail` is more than twice in comparison to the original model. Other features such as the age (`age_ship`) and the total thyroid volume (`sd_volg`) become more important.

Longitudinal Information

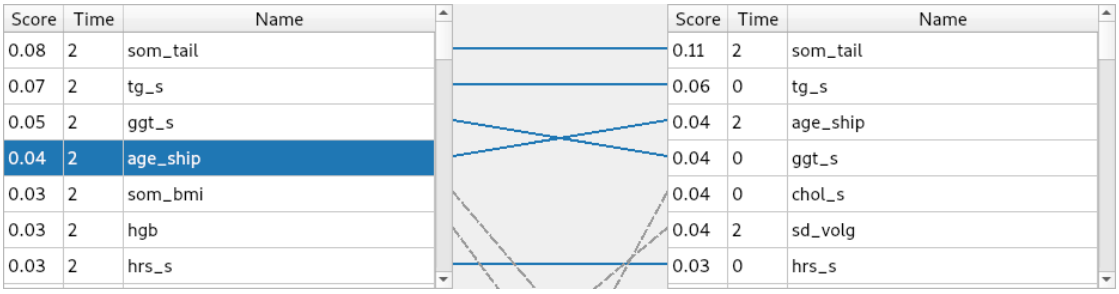
Apparently, the information given in serum biomarkers can not be obtained from other attributes. Maybe it is however sufficient to use the attribute values of a *previous* examination, which could save time and expenses. Thus, another classifier is learned, where the values for blood serum are taken from SHIP-0. The remaining attributes still use the latest examination values (i.e. SHIP-2). One could assume



Score	Time	Name
0.08	2	som_tail
0.07	2	tg_s
0.05	2	ggt_s
0.04	2	age_ship
0.03	2	som_bmi
0.03	2	hgb
0.03	2	hrs_s
0.03	2	gluc_s

Score	Time	Name
0.17	2	som_tail
0.06	2	age_ship
0.05	2	sd_volg
0.04	2	alkligt
0.04	2	som_huef
0.04	2	hba1c
0.03	2	quick
0.03	2	fib_cl

Figure 6.4: Comparison of importance scores. Left: Blood serum biomarkers are included in the data. Right: Blood serum biomarkers are excluded from the data.



Score	Time	Name
0.08	2	som_tail
0.07	2	tg_s
0.05	2	ggt_s
0.04	2	age_ship
0.03	2	som_bmi
0.03	2	hgb
0.03	2	hrs_s

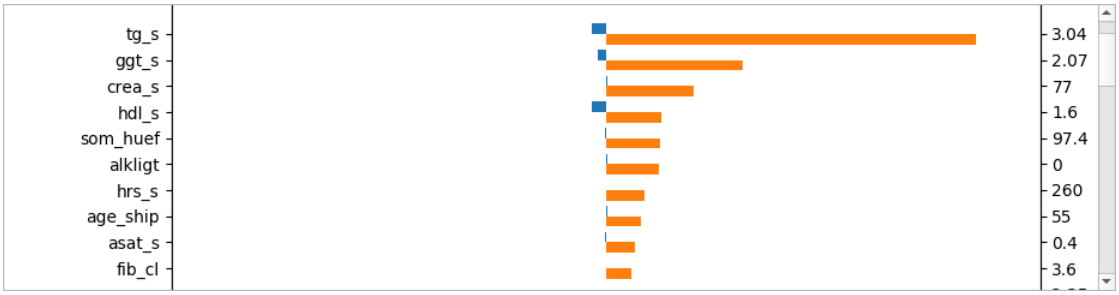
Score	Time	Name
0.11	2	som_tail
0.06	0	tg_s
0.04	2	age_ship
0.04	0	ggt_s
0.04	0	chol_s
0.04	2	sd_volg
0.03	0	hrs_s

Figure 6.5: Comparison of importance scores. Left: The most recent values for blood serum biomarkers are used. Right: The values of SHIP-0 are used for blood serum biomarkers.

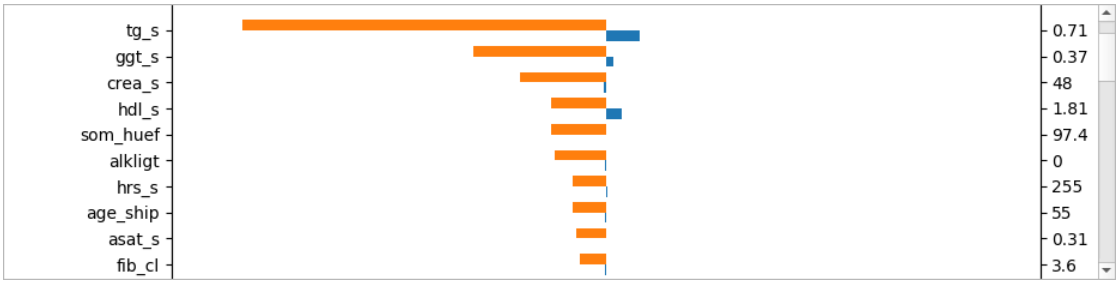
that the classification quality is worse than using the most recent values, but better than not having the values at all. One could also assume that old blood serum information is less important to the classifier than more recent laboratory results.

These assumptions can not be confirmed in the evaluation. Using the SHIP-0 values for all serum attributes reduces the accuracy to 0.72 and kappa score to 0.31. The most important features did not change much (see Figure 6.5); `som_tail` however gained importance. A more detailed evaluation of the differences in the global importances, as well as between the partial dependences, is omitted for sake of brevity. Instead, the trained classification models are exemplarily used to investigate differences on an individual level:

Participant 637 does not have hepatic steatosis. This has been correctly predicted by the original classifier (using the latest blood serum values), but is incorrectly predicted by the second classifier (using the blood serum values of SHIP-0), which classifies the woman as belonging to class 2, i.e. having a high liver fat percentage. Using the explainVis plot, this difference can be explained: The second classifier



(a) Old blood serum values



(b) Recent blood serum values

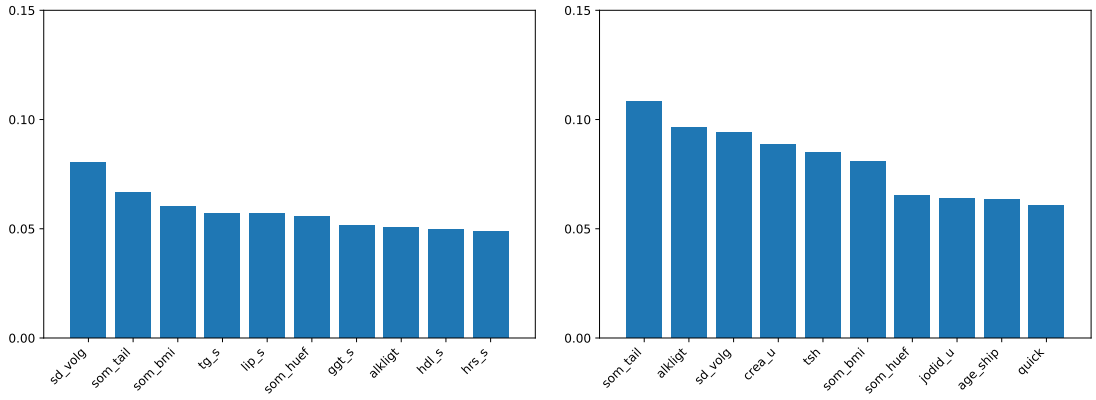
Figure 6.6: Comparison of feature contributions towards class 2 for participant 637. (a): Old blood serum values. (b): Recent blood serum values.

justifies its prediction with the participant's high triglyceride and gamma-GT levels in SHIP-0 (see Figure 6.6(a)). These levels however changed over time, and are in SHIP-2 at a lower, normal level (see Figure 6.6(b)). Here, one could argue that only using serum levels from ten years ago is not sufficient to classify, whether the participant has hepatic steatosis today.

6.2.4 Comparison with Internal Interpretation

The classification models presented in Section 2.2.2 and implemented in the framework all offer – with exception of the neural network – some internal interpretation, i.e. interpretation methods inherently contained due to the model design. A comparison of the interface's findings to the findings of the internal interpretation allows for an additional validation. Furthermore, the internal interpretation is likely applied in a real-world settings, where no sophisticated model interpretation is used. It may be therefore regarded as the "status quo".

The model used in the evaluation of the previous section was gradient boosting, a tree-based ensemble classifier. Ensembles are more difficult to interpret than single models, as there are many classifiers in the "black box", all predicting independently. However, multiple strategies have been developed to rank the features based on



(a) All features **(b)** No blood serum biomarkers

Figure 6.7: Ten most important features in $SHIP_W$, as computed internally by the gradient boosting classifier.

their importance to the boosting ensemble[1]. For one, it can be measured how often a variable has been selected for splitting in a tree, or how much it reduced the impurity. These values are then averaged over all trees. For another, one can use the data not used for building a tree (*out-of-bag samples*) as a test set for said tree. The importance of a predictor is then the prediction difference between the tree and a randomly permuted tree, tested on the out-of-bag samples [15].

XGBoost offers several of these method to assess the feature importance. Here, the relative number of times a feature is used for splitting is computed. Figure 6.7(a) shows the ten most important features in $SHIP_W$, Figure 6.7(b) shows the ten most important features, if blood serum biomarkers are removed. As one can see, roughly the same attributes are considered as important. Several differences exist however: For example, `sd_volg` has not been considered as a meaningful attribute by the framework, but is estimated as the most important attribute by the classifier. Notable is also `som_tail`: The proposed interface assigns it an importance score of 0.17, if no blood serum features are used. This is almost thrice the importance of the second-most import feature. The internal interpretation also assesses `som_tail` as the most important feature; However, the difference to the second-most important feature is much smaller.

[1]The following explanations also apply to other tree-based ensembles, for instance random forest.

6.3 Case Study: Synthetic Data

In the previous section, the proposed interface has been evaluated towards its usefulness to answer a given research question. Another application of the framework is its ability for *exploration*, i.e. generation of new hypotheses. Here, an evaluation on SHIP would be exhausting, due to its size and heterogeneity. Instead, a synthetic data set will be introduced. The evaluation task will be to get a general understanding of the influence of the features, and formulate new hypotheses if possible.

To simulate an authentic evaluation setting, the data set has been created by a scientist who is not familiar with the characteristics of the visual interface and all of its functions. This avoids the creation of data specifically fitted to the framework's properties. In contrast, the evaluator is familiar with the visual interface, but has no prior knowledge about the data.

Similar to the case study on SHIP, the following evaluation first gives a description of the data, then the evaluation steps are presented.

6.3.1 Data Description

The data set consists of 1000 observations and 10 attributes named `V1` to `V10`. `V10` stores the binary class information, i.e. is either 0 or 1. As a convenience it is renamed to `Label`. The remaining attributes are numerical. Other information, such as relations between the attributes, is not known. The data is almost perfectly balanced with respect to the class attribute: 512 instances are labelled as positive (`Label` = 1) and 488 as negative (`Label` = 0). Each feature contains only one value (in contrast to cohort studies, where multiple values exist due to repeated examinations).

6.3.2 Evaluation

General Classification Quality

Equivalently to the evaluation of $SHIP_W$, the first step in this evaluation is to train all classifiers on the data set, to analyse which classifiers perform well and are therefore worth interpreting. As can be seen in Table 6.4, the tree-based models can be trained perfectly on the data and predict the test instances without any error. The logistic regression and neural network are almost perfect, too, and fail on few instances only.

For the remainder of this evaluation, the decision tree is used as the selected classification model, due to its perfect score and simplicity.

Table 6.4: Classification Results on the synthetic data

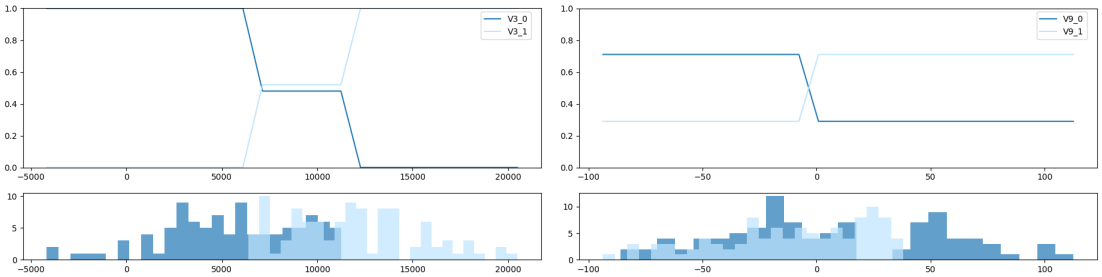
Model	Accuracy	Kappa
Decision Tree	**1**	**1**
Random Forest	**1**	**1**
Gradient Boosting	**1**	**1**
Logistic Regression	0.98	0.96
Neural Network	0.97	0.95

Score	Time	Name
0.50	0	V3
0.21	0	V9
0.00	0	V1
0.00	0	V2
0.00	0	V4

Figure 6.8: Global feature importance scores

Feature Importance and Partial Dependence

In Figure 6.8, the resulting global feature importance scores are displayed. Apparently, the classifier only uses two features in its prediction process: `V3` and `V9`. To understand, in which way these features influence the model, one can use the partial dependence plots. Figure 6.9 shows the PDPs for the two relevant features. Here, it becomes obvious that `V3` influences the prediction in the following way: `V3` $< 7000 \rightarrow$ `Label` $= 0$; `V3` $> 12,000 \rightarrow$ `Label` $= 1$. In the interval [7000,12000], `V3` does not influence the prediction. For `V9` can be said that a positive classification is more likely, if it is at least 0.



(a) `V3` **(b)** `V9`

Figure 6.9: Partial dependence plots of `V3` and `V9`. All classes are displayed.

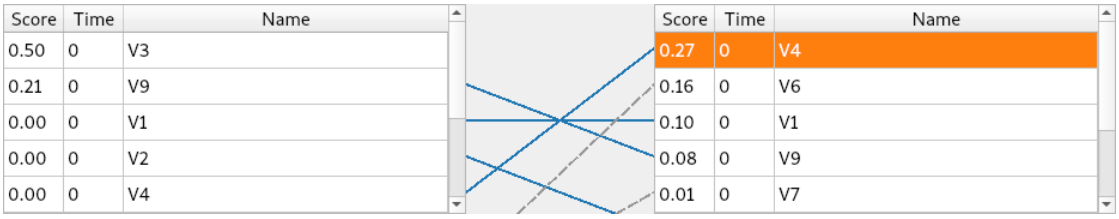


Figure 6.10: Comparison of importance scores. Left: All features are used. Right: `V3` has been removed.

Feature Combinations

The SilverEye panel allows to analyse, whether any feature combinations are exploited by the classifier. This is – in contrast to the evaluation of $SHIP_W$ – doable in reasonable time. If the classifier however only utilises two features, there can be only one feature combination between `V3` and `V9`, which has been accordingly detected by SilverEye.

Comparing Classification Models

In a next step, it is explored whether the decision tree can be properly learned on a subset of attributes, e.g. if some attributes are removed. First, `V3` as the most influential feature is removed. With a resulting accuracy of 0.94 and a kappa of 0.87, the model is still very predictive, but can not fully compensate the loss of `V3`. The global feature importance scores reveal, that a number of other attributes is used to replace `V3`, mainly `V4`, `V6` and `V1`. `V9` actually lost importance (see Figure 6.10). SilverEye reveals, that `V4` interacts with `V1`, but only barely with `V6` or `V9`. `V6` on the other hand does not interact with any other attribute (see Figure 6.11).

The apparent interaction between `V1` and `V4` can be inspected by browsing through the individuals. Here, one may notice that the two features are usually not used together. Instead, `V1` is often relevant to a prediction if `V4` is *not* relevant. Thus, it can be assumed that they somehow complement each other. Respective visualisations are omitted for sake of brevity.

By removing `V9` (but keeping `V3` this time), the classifier does not worsen, but fully compensates the attribute loss of `V9` with `V1`. The classification accuracy and kappa score stay at 1. Apparently, these features are highly correlated and contain the same amount of information, which is why only one of them is needed for classification.

6.3.3 Comparison with Internal Interpretation

Equivalently to the first case study, a comparison to the internal interpretation of the evaluated classifier is drawn. The classification model used in the evaluation is

Attribute	Goodness (Model 1)	Goodness (Model 2)
V4	0.7812	0.842
V1	0.7876	0.9484
V2	0.7876	0.8452
V3	0.7852	not present
V5	0.7852	0.8452
V6	0.7852	0.852
V7	0.7852	0.8436
V8	0.7928	0.8436
V9	0.7908	0.8708
V6	0.7816	0.842
V1	0.7876	0.8384
V2	0.7876	0.8428
V3	0.7852	not present
V4	0.7852	0.854
V5	0.7852	0.844
V7	0.7852	0.844
V8	0.7928	0.844
V9	0.7908	0.8444

Figure 6.11: SilverEye scores for `V4` and `V6`. Left: All features are used. Right: `V3` has been removed.

a decision tree. This model type can be interpreted by drawing the inducted tree as a graph. Here, the expert can see, which features have been used for splitting, and which consequences a split has. Depending on the classification problem and used regularisation, trees may overfit and grow very deep. In this case, an interpretation becomes unintuitive.

This problem does not exist in the simple synthetic data set. The inducted tree is shown in Figure 6.12. One can clearly validate the findings of the previous section: `V3` perfectly classifies all instances outside the interval of [6297, 11268]; The remaining instances are classified via `V9`.

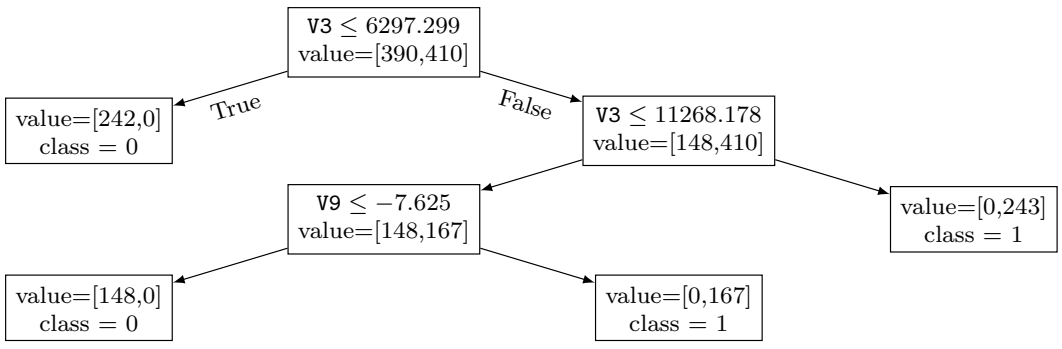


Figure 6.12: Decision tree trained on the synthetic data set.

6.4 Discussion

The evaluation showed, that the proposed interface is able to analyse the relevance and influence of features in a data set. The first case study tested, whether the framework works on a real-world epidemiological cohort study and can answer a research question. This can be confirmed. The second case study tested, whether the framework can be used for a general evaluation of a data set. This can be confirmed, too. In comparison to the interpretation techniques internally contained in the classifiers, the visual interface offered much more support and insight into the classifier's reasoning.

The global feature importance scores allowed to give a first insight into which features are relevant at all. It would have been useful, if the scores are put in relation to the maximum achievable importance (which depends on the original prediction probability). The bipartite graph was useful to compare the rankings.

Of particular usefulness has been the individual analysis, despite not being directly focus of the research question. Understanding, why a classification model has assigned a label to a specific participant, is insightful and promising for a productive application. The filtering mechanism allowed to quickly navigate to certain instances, for example incorrectly classified instances. The individual analysis was especially useful in conjunction with the partial dependence panel: Here, the evaluator can compare individual's feature to the feature distribution and global prediction probability. A link between these panels, which synchronises the selected class and highlights the feature, would have been advantageous.

The partial dependence plot and the distribution histogram have been useful, too. They allowed to understand the influence of a feature to the prediction probability, while showing the data distribution. In the first case study, outliers restricted the partial dependence plots, because they caused parts of the plot to be uninformative. Here, more preprocessing in terms of outlier removal would have been useful. For the second case study, it would have been also beneficial if the partial dependence computation can be restricted to a subpopulation.

SilverEye has not been of much usefulness. In the first case study, it could not be used due to the high-dimensionality of the data. Here, feature selection would have been of advantage. On the synthetic data set, SilverEye detected existing combinations between features. However, the detected combinations were mostly expected, as they only revealed the overall important attributes. The original idea – finding features which are only meaningful in combination – could not be confirmed. Problematic is in particular the influence of randomness: If goodness values fluctuate despite being based on the same grouping, it is hard to estimate which grouping actually worked well.

The incorporation of longitudinal information is somewhat limited, too. By replacing the value of a feature with a previous record, the user can compare the importances of two models. The effectiveness of this has been shown in the first case study. This process takes however time, if the user wants to explore the data instead of only validating existing research questions.

7. Conclusion

7.1 Summary

Epidemiology studies the cause and spread of diseases by collecting and analysing health-related conditions in human populations. In epidemiological cohort studies, population members are monitored over a time span and repeatedly examined, allowing to track the development of health conditions. As the standard workflow to analyse such population studies is challenged by increasing data sizes, researchers successfully started the incorporation of data mining into their analyses, most notable classification algorithms. State-of-the-art classifiers often lack interpretability – an important desideratum for medical experts.

This thesis attempts to support epidemiologists in understanding trained classification models. Five requirements have been identified, which specifically address the characteristics of cohort studies: The support of multivariate and heterogeneous data, the independence towards the used classification model including multi-class scenarios, the consideration of longitudinal features, and the aggregation into an interactive visual interface.

As no existing work satisfies all of these requirements, a novel visual interface has been developed. It supports any tabular data, as well as any probabilistic classification model. Several methods are integrated in the interface, allowing for a model interpretation at different levels. The first key component is to determine, which features of an individual have contributed to its classification. This is computed by observing the change in prediction, if the feature is averaged out. The resulting importances are displayed in a bar chart and compared to the average importance. To get a comparable global feature importance score, the individual

scores are averaged over all instances and classes. The relation between the value of a feature and the prediction probability is analysed by the partial dependence, which is displayed in an enhanced plot. A novel technique called SilverEye has been developed, which allows to successively construct meaningful feature groups by measuring, how their mutual random permutation affects the classification. Another key concept is the comparison of two trained classifiers, which are learned on different feature sets. A feature set can be modified by removing features or replacing a feature's value by a previous record. Thus, the user can interpret how a classifier behaves if no recent information about a feature is available. The ability for comparison is present in all components of the framework.

The proposed interface has been evaluated with respect to its effectiveness for interpreting classification models in an epidemiological context. Here, two case studies have been presented and performed. In the first case study, the evaluator had to validate a research question on SHIP, a real-world cohort study. The research question required the usage of virtually all interface components, including a comparison of multiple models in order to evaluate the influence of previous examination recordings. Due to the large size of the SHIP data, a second case study was performed on a smaller, synthetic data set. Here, the ability of the visual interface for exploring new, unknown data was evaluated. No specific research question has been given; Instead, the evaluator was asked to gain a general understanding of the features and their influence in the classifier.

The results of the evaluation confirm the usefulness of the interface for the desired tasks. Both case studies discovered insights into the classifier's reasoning, but also revealed the potential and necessity of further work. For one, not all components have been relevant in the evaluation, and should be improved. Also, the components could interact more with each other, allowing for simpler interpretations. To be employed in a practical application, the visual interface needs to be adjusted and tested against the requirements of epidemiological workflows.

7.2 Future Work

The work discussed in this thesis forms a basis for future research.

One aspect addresses the components of the interface. Here, several modifications may be considered due to the evaluation results. For one, SilverEye has not been very useful in the case studies and should be either redesigned, or removed in favour of other strategies. One could also consider the usage of more sophisticated visualisations in the multi-class scenario. The current solution – combo boxes in the partial dependence and individual panel – does not always allow a direct comparison between the classes. In general, the framework may be further developed to allow

more linkage and interaction between the components. As already mentioned, one could for example synchronise the inspected class in the individual and PD panel.

The components in the interface have been carefully selected with respect to the stated requirements and their diversity. Here, one could also implement and evaluate more ideas. Krause et al. [50] for example allow the user to modify an individual's feature value and observe, if the classifier reacts to the modification. With respect to the epidemiological context, this idea is promising.

The aspect of longitudinal features has been addressed in the framework in the ability of using old examination values. Therefore, only one value per feature can be included in the classification process. Future work could discuss, how to incorporate and interpret feature sequences, where all previous recordings are included.

Another aspect concerns the adjustment of the framework to fit into an epidemiological workflow. Cibulski [18] defined several requirements for visual analytics in this area. Here, functionality such as a history log with undo operation, a text editor or the storage of preferred settings is relevant and should be added to the interface. In order to evaluate this aspect, a usability study with an epidemiologist should be performed. His feedback will likely give much insight into the expert's point of view.

The model-agnostic property of the visual interface allows the integration into standard data mining workflows. Here, the framework could benefit from previous steps in the workflow such as preprocessing or feature selection. In addition, model interpretation may also contribute to the feedback loop in interactive machine learning.

The ability of the framework to replace the value of an attribute with a related value is motivated by epidemiological cohort studies, where participants are repeatedly examined. However, this concept may be also applied to other areas utilising time-related or otherwise related features, for example the estimation of a company's liquidity by using their financial history.

A. Appendix

Table A.1: Attributes in SHIP

Attribute Code	Unit	Description
age_ship	Years	Age at examination day
diabetes	No/Yes	Has diabetes
smoking	0/1/2	Never smoked / has smoked / smoking
alkligt	g	Monthly ethanol intake
abstain	No/Yes	Abstinence from alcohol (12 months)
hyperlipid	No/Yes	Treated hyperlipidemia
gout	No/Yes	Treated gout
som_bmi	kg/m^2	Body mass index
som_tail	cm	Waist circumference
som_huef	cm	Hip size
hgb	g/l	Haemoglobin
hba1c	%	Glycated haemoglobin A1c
quick	%	Thromboplastin time Quick
fib_cl	g/l	Fibrinogen (Clauss)
crea_s	µmol/l	Serum creatinine
hrs_s	µmol/l	Serum uric acid
gluc_s	mmol/l	Serum glucose
asat_s	µmol/sl	Serum ASAT
ggt_s	µmol/sl	Serum GGT
lip_s	µmol/sl	Serum lipase
chol_s	mmol/l	Serum cholesterol

Continued on next page

Table A.1 – *Continued from previous page*

Attribute Code	Unit	Description
tg_s	mmol/l	Serum triglycerides
hdl_s	mmol/l	Serum HDL
ldl_s	mmol/l	Serum LDL
tsh	mu/l	Thyroid-stimulating hormone
jodid_u	µg/dl	Iodide (urine)
crea_u	mmol/l	Creatinine (urine)
sd_volg	g	Total thyroid volume
goiter	No/Yes	Goiter
knoten	No/Yes	At least one thyroid node
earm	0/1	Echonormal/abnormal thyroid pattern
atc_c07a	No/Yes	Beta blocking agents
atc_c07aa	No/Yes	Beta blocking agents, non-selective
atc_c07ab	No/Yes	Beta blocking agents, selective
atc_c08	No/Yes	Calcium channel blockers
atc_c08ca01	No/Yes	Amlodipine
atc_c08ca05	No/Yes	Nifedipine
atc_c08ca08	No/Yes	Nitrendipine
atc_c08da01	No/Yes	Verapamil
atc_c09aa02	No/Yes	Enalapril
atc_c09aa05	No/Yes	Ramipril
atc_h02a	No/Yes	Corticosteroids for systemic use
		$SHIP_W$ specific
hormonrepl_w	No/Yes	Hormone replacement therapy
menopaus_w	Years	Age at entry into menopause
menopause_yn_w	No/Yes	Natural menopause
parity_w	No/Yes	Parity, at least one pregnancy

Bibliography

[1] Philip Adler, Casey Falk, Sorelle A. Friedler, Gabriel Rybeck, Carlos Scheidegger, Brandon Smith, and Suresh Venkatasubramanian. Auditing Black-box Models for Indirect Influence. *Knowledge and Information Systems*, 54(1): 95–122, 2018. (cited on Page 23)

[2] Shiva Alemzadeh, Tommy Hielscher, Uli Niemann, Lena Cibulski, Till Ittermann, Henry Völzke, Myra Spiliopoulou, and Bernhard Preim. Subpopulation Discovery and Validation in Epidemiological Data. *European conference on computer vision workshop on Visual Analytics*, (June):2–6, 2017. (cited on Page 52)

[3] Saleema Amershi, Maya Cakmak, William Bradley Knox, and Todd Kulesza. Power to the People: The Role of Humans in Interactive Machine Learning. *AI Magazine*, 35(4):105, 2014. (cited on Page 37)

[4] Saleema Amershi, Max Chickering, Steven M. Drucker, Bongshin Lee, Patrice Simard, and Jina Suh. ModelTracker: Redesigning Performance Analysis Tools for Machine Learning. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 337–346. ACM, 2015. (cited on Page ix and 29)

[5] Andrew C. Lorenc. Analysis methods for numerical weather prediction. *Quarterly Journal of the Royal Meteorological Society*, 112(474):1177–1194, 1986. (cited on Page 14)

[6] Robert Andrews, Joachim Diederich, and Alan B. Tickle. Survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowledge-Based Systems*, 8(6):373–389, 1995. (cited on Page 27)

[7] Daniel W Apley. Visualizing the Effects of Predictor Variables in Black Box Supervised Learning Models. pages 1–36. (cited on Page 22)

[8] David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Mueller. How to Explain Individual Classification Decisions. *Journal of Machine Learning Research*, 11:1803–1831, 2010. (cited on Page 28)

[9] Osbert Bastani, Carolyn Kim, and Hamsa Bastani. Interpreting Blackbox Models via Model Extraction. 2017. (cited on Page 27)

[10] Giorgio Bedogni, Stefano Bellentani, Lucia Miglioli, Flora Masutti, Marilena Passalacqua, Anna Castiglione, and Claudio Tiribelli. The Fatty Liver Index: A simple and accurate predictor of hepatic steatosis in the general population. *BMC Gastroenterology*, 6:1–7, 2006. (cited on Page 50 and 54)

[11] Michael Behrisch, James Davey, Svenja Simon, Tobias Schreck, Daniel Keim, and Jörn Kohlhammer. Visual Comparison of Orderings and Rankings. In *EuroVis Workshop on Visual Analytics*, pages 1–5, 2013. (cited on Page 41)

[12] Jacob Bien and Robert Tibshirani. Prototype Selection for Interpretable Classification. *Annals of Applied Statistics*, 5(4):2403–2424, 2011. (cited on Page 28)

[13] Or Biran and Courtenay Cotton. Explanation and Justification in Machine Learning: A Survey. In *IJCAI-17 Workshop on Explainable AI*, pages 8–13, Melbourne, 2017. (cited on Page 17)

[14] Leo Breiman. Bagging Predictors. *Machine Learning*, 24:123–140, 1996. (cited on Page 11)

[15] Leo Breiman. Random Forests. *Machine Learning*, 45:5–32, 2001. (cited on Page 59)

[16] Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. *Classification and Regression Trees*. Routledge, New York, 1 edition, 1984. (cited on Page 48)

[17] Anna L Buczak, Phillip T Koshute, Steven M Babin, Brian H Feighner, and Sheryl H Lewis. A data-driven epidemiological prediction method for dengue outbreaks using local and remote sensing data. *BMC Medical Informatics and Decision Making*, 12(1):124, 2012. (cited on Page 8)

[18] Lena Cibulski. Visual Analytics Support for Analysis of Cohort Study Data : Requirements and Concepts. Technical report, Otto-von-Guericke-University, Magdeburg, 2016. (cited on Page 69)

[19] Paulo Cortez and Mark J Embrechts. Using Sensitivity Analysis and Visualization Techniques to Open Black Box Data Mining Models. *Information Sciences*, 225:1–17, 2013. (cited on Page 23)

[20] Paulo Cortez and Mark J. Embrechts. Opening Black Box Data Mining Models Using Sensitivity Analysis. In *2011 IEEE Symposium on Computational Intelligence and Data Mining*, pages 341–348. IEEE, 2013. (cited on Page 23)

[21] Padraig Cunningham, Donal Doyle, and John Loughrey. An Evaluation of the Usefulness of Case-Based Explanation. In *International Conference on Case-Based Reasoning*, pages 122–130, Trondheim, 2003. Springer. (cited on Page 28)

[22] Piotr Dabkowski and Yarin Gal. Real Time Image Saliency for Black Box Classifiers. In *31st Conference on Neural Information Processing Systems*, pages 6970–6979, Long Beach, CA, USA, 2017. (cited on Page 20)

[23] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. SAGA: A Fast Incremental Gradient Method With Support for Non-Strongly Convex Composite Objectives. In *Advances In Neural Information Processing Systems*, 2014. (cited on Page 49)

[24] Finale Doshi-Velez and Been Kim. Towards A Rigorous Science of Interpretable Machine Learning. 2017. (cited on Page 18)

[25] Wouter Duivesteijn and Julia Thaele. Understanding Where Your Classifier Does (Not) Work – The SCaPE Model Class for EMM. In *IEEE International Conference on Data Mining*, pages 809–814, 2014. (cited on Page 28)

[26] John Ehrlinger. ggRandomForests: Visually Exploring a Random Forest for Regression. *arXiv preprint*, 1501.07196, 2016. (cited on Page 22)

[27] Mark J. Embrechts, Fabio A. Arciniegas, Muhsin Ozdemir, and Robert H. Kewley. Data Mining for Molecules with 2-D Neural Network Sensitivity Analysis. *International Journal of Smart Engineering System Design*, 5(4): 225–239, 2003. (cited on Page 23)

[28] Gary D. Friedman. *Primer of Epidemiology*. McGraw-Hill, New York, 1974. (cited on Page 5)

[29] Jerome H. Friedman. Greedy Function Approximation: A Gradient Boosting Machine. *The Annals of Statistics*, 29(5):1189–1232, 2001. (cited on Page 12, 21, and 43)

[30] Matthieu Geist. Soft-max boosting. *Machine Learning*, 100(2-3):305–332, 2015. (cited on Page 49)

[31] Alex Goldstein, Adam Kapelner, Justin Bleich, and Emil Pitkin. Peeking Inside the Black Box: Visualizing Statistical Learning With Plots of Individual Conditional Expectation. *Journal of Computational and Graphical Statistics*, 24(1):44–65, 2015. (cited on Page 22)

[32] Riccardo Guidotti, Anna Monreale, Franco Turini, Dino Pedreschi, and Fosca Giannotti. A Survey Of Methods For Explaining Black Box Models. *arXiv preprint*, 1802.01933, 2018. (cited on Page 18 and 23)

[33] Jiawei Han and Micheline Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers Inc., 2000. (cited on Page 14)

[34] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer, New York, 2011. (cited on Page 11 and 12)

[35] Andreas Henelius, Kai Puolamäki, Henrik Boström, Lars Asker, and Panagiotis Papapetrou. A peek into the black box: exploring classifiers by randomization. *Data Mining and Knowledge Discovery*, 28(5-6):1503–1529, 2014. (cited on Page xi, 24, 45, and 46)

[36] Andreas Henelius, Kai Puolamäki, Isak Karlsson, Jing Zhao, Lars Asker, Henrik Boström, and Panagiotis Papapetrou. GoldenEye++: A Closer Look into the Black Box. In *International Symposium on Statistical Learning and Data Sciences*, pages 96–105. Springer, 2015. (cited on Page ix, 25, 26, and 46)

[37] Tommy Hielscher, Myra Spiliopoulou, Henry Völzke, and Jens Peter Kühn. Using Participant Similarity for the Classification of Epidemiological Data on Hepatic Steatosis. In *IEEE Symposium on Computer-Based Medical Systems (CBMS)*, pages 1–7. IEEE, 2014. (cited on Page 52 and 53)

[38] Tommy Hielscher, Myra Spiliopoulou, Henry Völzke, and Jens-Peter Kühn. Mining Longitudinal Epidemiological Data to Understand a Reversible Disorder. In *International Symposium on Intelligent Data Analysis*, pages 120–130. Springer, 2014. (cited on Page 51 and 53)

[39] David W. Hosmer and Stanley Lemeshow. *Applied Logistic Regression*. Wiley, 2nd edition, 2000. (cited on Page 12 and 14)

[40] Minsuk Kahng, Dezhi Fang, and Duen Horng Chau. Visual Exploration of Machine Learning Results using Data Cube Analysis. In *Proceedings of the*

Workshop on Human-In-the-Loop Data Analytics - HILDA '16, pages 1–6. ACM, 2016. (cited on Page ix, 31, and 32)

[41] Minsuk Kahng, Pierre Y. Andrews, Aditya Kalro, and Duen Horng Chau. ActiVis: Visual Exploration of Industry-Scale Deep Neural Network Models. *IEEE transactions on visualization and computer graphics*, 24(1):88–97, 2018. (cited on Page 29)

[42] Ioannis Kavakiotis, Olga Tsave, Athanasios Salifoglou, Nicos Maglaveras, Ioannis Vlahavas, and Ioanna Chouvarda. Machine Learning and Data Mining Methods in Diabetes Research. *Computational and Structural Biotechnology Journal*, 15:104–116, 2017. (cited on Page 15 and 16)

[43] Daniel Keim, Jörn Kohlhammer, Geoffrey Ellis, and Florian Mansmann. *Mastering the Information Age Solving Problems with Visual Analytics*. Eurographics Association, 2010. (cited on Page 47)

[44] Been Kim, Cynthia Rudin, and Julie Shah. The Bayesian Case Model: A Generative Approach for Case-Based Reasoning and Prototype Classification. *Advances In Neural Information Processing Systems*, pages 1–9, 2014. (cited on Page 28)

[45] Been Kim, Rajiv Khanna, and Oluwasanmi Koyejo. Examples are not Enough, Learn to Criticize! Criticism for Interpretability. *Advances In Neural Information Processing Systems*, pages 2280–2288, 2016. (cited on Page 28)

[46] Jaedeok Kim and Jingoo Seo. Human Understandable Explanation Extraction for Black-box Classification Models Based on Matrix Factorization. *arXiv preprint*, 1709.06201, 2017. (cited on Page 27)

[47] Diederik P. Kingma and Jimmy Lei Ba. Adam: a Method for Stochastic Optimization. *International Conference on Learning Representations 2015*, pages 1–15, 2015. (cited on Page 49)

[48] David G. Kleinbaum, Lawrence L. Kupper, and Lloyd E. Chambless. Logistic regression analysis of epidemiologic data: Theory and practice. *Communications in Statistics - Theory and Methods*, 11(5):485–547, 1982. (cited on Page 12)

[49] Josua Krause, Adam Perer, and Enrico Bertini. Using Visual Analytics to Interpret Predictive Machine Learning Models. *arXiv preprint*, 1606.05685, 2016. (cited on Page 16 and 22)

[50] Josua Krause, Adam Perer, and Kenney Ng. Interacting with Predictions: Visual Inspection of Black-box Machine Learning Models. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 5686–5697. ACM, 2016. (cited on Page ix, 22, 29, 31, 43, and 69)

[51] Josua Krause, Aritra Dasgupta, Jordan Swartz, Yindalon Aphinyanaphongs, and Enrico Bertini. A Workflow for Visual Diagnostics of Binary Classifiers using Instance-Level Explanations. *arXiv preprint*, 1705.01968, 2017. (cited on Page ix, 30, and 32)

[52] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances In Neural Information Processing Systems*, pages 1097–1105, 2012. (cited on Page 17)

[53] Rudolf Kruse, Christian Borgelt, Christian Braune, Sanaz Mostaghim, and Matthias Steinbrecher. *Computational Intelligence*. Springer, London, 2nd edition, 2016. (cited on Page 13)

[54] Max Kuhn and Kjell Johnson. *Applied Predictive Modeling*. Springer, New York, 5 edition, 2016. (cited on Page 9 and 17)

[55] Todd Kulesza, Margaret Burnett, Weng-Keen Wong, and Simone Stumpf. Principles of Explanatory Debugging to Personalize Interactive Machine Learning. In *Proceedings of the 20th International Conference on Intelligent User Interfaces - IUI '15*, pages 126–137. ACM, 2015. (cited on Page ix, 29, and 30)

[56] J Richard Landis and Gary G Koch. The Measurement of Observer Agreement for Categorical Data. *Biometrics*, 33(1):159–174, 1977. (cited on Page 53)

[57] Vincent Lemaire, Raphael Féraud, and Nicolas Voisine. Contact Personalization using a Score Understanding Method. In *Proceedings of the International Joint Conference on Neural Networks*, pages 649–654. IEEE, 2008. (cited on Page 23)

[58] Jiuyong Li, Ada Wai-chee Fu, and Paul Fahey. Efficient discovery of risk patterns in medical data. *Artificial Intelligence in Medicine*, 45(1):77–89, 2009. (cited on Page 8)

[59] Zachary C. Lipton. The Mythos of Model Interpretability. *arXiv preprint*, 1606.03490, 2016. (cited on Page 18 and 19)

[60] Scott Lundberg and Su-In Lee. A Unified Approach to Interpreting Model Predictions. In *31st Conference on Neural Information Processing Systems*, 2017. (cited on Page 24)

[61] Shannon M Lynch and Jason H Moore. A call for biological data mining approaches in epidemiology. *BioData Mining*, 9(1):2015–2017, 2016. (cited on Page 8)

[62] D Martens, B Baesens, T. Van Gestel, and J. Vanthienen. Comprehensible Credit Scoring Models Using Rule Extraction From Support Vector Machines. *European Journal of Operational Research*, 183(13):1466–1476, 2007. (cited on Page 27)

[63] David Martens and Foster Provost. Explaining Data-Driven Document Classifications. 2014. (cited on Page 20)

[64] Benedikt Mayer. Visual Analytics of Participant Evolution in Longitudinal Cohort Study Data. Technical report, Otto-von-Guericke-University, Magdeburg, 2018. (cited on Page 51)

[65] Mary L. McHugh. Interrater reliability: the kappa statistic. *Biochemia Medica*, pages 276–282, 2012. (cited on Page 10)

[66] Tim Miller. Explanation in Artificial Intelligence: Insights from the Social Sciences. *arXiv preprint*, 1706.07269, 2017. (cited on Page 16)

[67] Uli Niemann, Henry Völzke, Jens Peter Kühn, and Myra Spiliopoulou. Learning and inspecting classification rules from longitudinal epidemiological data to identify predictive features on hepatic steatosis. *Expert Systems with Applications*, 41(11):5405–5415, 2014. (cited on Page 51)

[68] Uli Niemann, Tommy Hielscher, Myra Spiliopoulou, Henry Völzke, and Jens Peter Kuhn. Can we classify the participants of a longitudinal epidemiological study from their previous evolution? In *28th International Symposium on Computer-Based Medical Systems*, pages 121–126. IEEE, 2015. (cited on Page 17 and 51)

[69] Stacy L. Özesmi and Uygar Özesmi. An artificial neural network approach to spatial habitat modelling with interspecific interaction. *Ecological Modelling*, 116(1):15–31, 1999. (cited on Page 20)

[70] Bernhard Preim, Paul Klemm, Helwig Hauser, Katrin Hegenscheid, Steffen Oeltze, Klaus Toennies, and Henry Völzke. Visual Analytics of Image-Centric Cohort Studies in Epidemiology. In *Visualization in Medicine and Life Sciences III*, pages 221–248. Springer, 2016. (cited on Page 5)

[71] Janardan K. Reddy and M. Sambasiva Rao. Lipid Metabolism and Liver Inflammation. II. Fatty liver disease and fatty acid oxidation. *American*

Journal Of Physiology Gastrointestinal Liver Physiology, pages G852–G858, 2006. (cited on Page 52)

[72] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144, New York, 2016. ACM. (cited on Page ix, 26, 27, and 40)

[73] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Anchors: High-Precision Model-Agnostic Explanations. In *AAAI Conference on Artificial Intelligence*. AAAI, 2018. (cited on Page 27)

[74] Marko Robnik-Šikonja and Igor Kononenko. Explaining Classifications for Individual Instances. *IEEE Transactions on Knowledge and Data Engineering*, 20(5):589–600, 2008. (cited on Page ix, 23, 25, 40, and 42)

[75] Andrea Saltelli. Sensitivity Analysis for Importance Assessment. *Risk Analysis*, 22(3):579–590, 2002. (cited on Page 23)

[76] David Silver, Aja Huang, . . . , Thore Graepel, and Demis Hassabis. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587): 484–489, 2016. (cited on Page 17)

[77] Erik Štrumbelj and Igor Kononenko. An Efficient Explanation of Individual Classifications using Game Theory. *Journal of Machine Learning Research*, 11:1–18, 2010. (cited on Page 24)

[78] Erik Štrumbelj and Igor Kononenko. Explaining prediction models and individual predictions with feature contributions. *Knowledge and Information Systems*, 41(3):647–665, 2014. (cited on Page 24)

[79] Erik Štrumbelj, Igor Kononenko, and Marko Robnik-Šikonja. Explaining instance classifications with interactions of subsets of feature values. *Data and Knowledge Engineering*, 68(10):886–904, 2009. (cited on Page 23 and 24)

[80] Gábor J. Székely, Maria L. Rizzo, and Nail K. Bakirov. Measuring and Testing Dependence by Correlation of Distances. *Annals of Statistics*, 35(6):2769–2794, 2007. (cited on Page 46)

[81] Paolo Tamagnini, Josua Krause, Aritra Dasgupta, and Enrico Bertini. Interpreting Black-Box Classifiers Using Instance-Level Visual Explanations. In *Proceedings of the 2nd Workshop on Human-In-the-Loop Data Analytics*, pages 1–6, 2017. (cited on Page 27 and 30)

[82] Panh-Ning Tan, Michael Steinbach, and Vipin Kumar. *Introduction to Data Mining*. Pearson, 2005. (cited on Page 9)

[83] Robert Tibshirani. Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996. (cited on Page 49)

[84] Edward Rolf Tufte. *Envisioning Information*. Graphics Press, Cheshire, 1990. (cited on Page 41)

[85] Ryan Turner. A Model Explanation System. In *IEEE International Workshop on Machine Learning for Signal Processing, MLSP*, pages 1–6, Salerno, 2016. IEEE. (cited on Page 27)

[86] Tyler J. Van Der Weele and Mirjam J. Knol. A Tutorial on Interaction. *Epidemiologic Methods*, 3(1):33–72, 2014. (cited on Page 14)

[87] Henry Völzke, Dietrich Alte, . . . , Ulrich John, and Wolfgang Hoffmann. Cohort profile: The study of health in Pomerania. *International Journal of Epidemiology*, 40(2):294–307, 2011. (cited on Page 50 and 51)

[88] Daniel S Weld and Gagan Bansal. Intelligible Artificial Intelligence. *arXiv preprint*, 1803.04263, 2018. (cited on Page 16)

[89] Alexander Zien, Nicole Krämer, Sören Sonnenburg, and Gunnar Rätsch. The Feature Importance Ranking Measure. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 694–709. Springer, 2009. (cited on Page 23)

I herewith assure that I wrote the present thesis independently, that the thesis has not been partially or fully submitted as graded academic work and that I have used no other means than the ones indicated. I have indicated all parts of the work in which sources are used according to their wording or to their meaning.

I am aware of the fact that violations of copyright can lead to injunctive relief and claims for damages of the author as well as a penalty by the law enforcement agency.

Magdeburg, 4$^{\text{th}}$ May 2018