Explain variable influence in black box models through pattern mining

Xiaoqi Ma
xiaoqi.ma@rwth-aachen.de
Matriculation number: 383420

Supervisor: Prof. Dr. Markus Strohmaier
Second Examiner: Prof. Dr. Bastian Leibe
Advisor: Dr. Florian Lemmerich

Chair of Computational Social Sciences and Humanities
RWTH Aachen Faculty of Mathematics, Computer Science and Natural Sciences
RWTH Aachen University

This thesis is submitted for the degree of
M.Sc. Media Informatics

Aachen, Germany
December 10, 2019

Contents

1 Introduction

1.1 Motivation

In the recent decades, machine learning fields have been studied extensively. Simply to elucidate, machine learning is a set of methods that are used to teach computers to perform different tasks without hard-coding instructions. It has attracted much attention due to its powerful application, especially in the "Big data" era. Thanks to the boosting computational power, machine learning algorithms can make use of large volumes of data to achieve numerous tasks which are not expected before. For instance, a myriad of classification or regression tasks could be solved efficiently by applying machine learning algorithms. A simple regression task could be predicting the weather temperature by using logistic regression based on historical data, and a more complicated task could look like language translation problem.

Since there are various kinds of machine learning models, a considerable barrier for human engineers is how to choose the right models for specific problems. Generally, concerning the evaluation of machine learning models, people tend to pay more attention on the model performance rather than the model interpretability. The model performance is definitely very fundamental to assess the model, which typically can be measured by metrics like accuracy, precision, recall and etc. Nevertheless, we should not neglect the importance of model interpretability, which shows "the degree for a human to understand model decisions and the ability to consistently predict the results"[1]. Therefore, One of the major topics to be investigated in machine learning field is interpretable machine learning. It is defined as the use of ML models for the extraction of relevant knowledge about domain relationships contained in data. [2]

Broadly speaking, machine learning models can be categorized as white box models and black box models judging from the model interpretability. White box models can be roughly considered as interpretable models, which maintain high model interpretability. Usually they contain simple structures, a limited number of model parameters, and most importantly, the decisions made by white box models are interpretable by human. For example, interpretable models include linear regression, logistic regression, and decision tree model. Those models are human-understandable since the prediction results could be interpreted by examining the model parameters. On the contrary, black box models usually have more complex structures and a substantial number of parameters which are not intrinsically understandable. Ensemble models or neural networks are normally regarded as black box models for the reason that decisions made by black box models cannot be understood by looking at their parameters, which is a major disadvantage for complex models. Typically, those complicated models can achieve better performance for the sake of less interpretability. However, proper model interpretability is crucial to provide explanations to the decisions made by the model and especially important for decision-makers. Besides, "right to explanation", meaning the right to be given an explanation for an output of automated algorithms was stated by General Data Protection Regulation(GDPR), which requires businesses to provide understandable justifications to

their users [3]. One scenario is that the bank manager is obligated to clarify reasons to the user about the loan rejection if requested.

Since white box models are intrinsically interpretable, a more challenging problem which arises in this domain is how to explain the black box models. In other words, it is of paramount importance to investigate methods to give reasonable explanations to model predictions. Recent theoretical developments have revealed that there are approaches to interpret black box models, which can be summarized as global interpretation methods and local interpretation methods with respect to different viewpoints. As the name suggests, the global interpretation focus on the global view of the input variables. However, it would be of special interest to investigate local interpretation methods, which are operated on the instance level. Concluded by Alvarez-Melis [4], those local interpretation methods can be roughly categorized as salience-based and perturbation-based approaches. The former method category is also known as gradient-based attribution methods, computing the partial derivatives of the output with respect the each input feature, e.g. Integrated Gradients [5][6]. In contrast, perturbation-based approaches first generate a bunch of neighborhood data points surrounding the instance to be explained, then calculate the contribution of each input features towards the output by fitting a local interpretable model, e.g. LIME [7].

1.2 Purpose of this thesis

The foremost problem we are facing is how to interpret the black box models. Undoubtedly, many interpretation methods have already come to the surface to facilitate model explanation, but they are not sufficient to deal with complex situations. The global interpretation methods give a too broad interpretation view while local interpretation methods may become too sensitive to reveal the underlying cause due to the excessive interpretation of the target instance. Indeed, the insight gained from a single instance map might be too brittle, and lead to a false sense of understanding.

Therefore, this thesis aims to develop an overarching framework to provide reasonable explanations for black box model predictions, given the urgent need to obtain decent justifications for decisions made by the algorithms. To our knowledge, many studies have yielded interpretation frameworks that are just applicable to one type of data or to a specific kind of black box model. By taking various types of data and black box models into account, it is not evident what is the best type of explanation metric and framework. We hence intend to build an interpretation framework which includes diverse explanation methods in order to align with the purpose and completeness of the targeted explanation.

Another main contribution proposed in this thesis is that we aim to combine the local interpretation methods with the pattern mining technique since it seems not sufficient to explain model predictions by merely applying local interpretation approaches. It is convincing that black box models seem to capture the "hidden patterns" in data as to achieve good performance when performing classification or regression tasks. Similarly, we introduce the subgroup discovery technique to

demonstrate the feasibility of discovering patterns that can facilitate us to explain black box models, i.e. identifying the patterns in data where a selected variable impose a significant influence. From data instance perspective, the interpretation level of this novel approach is somewhere between local view and global view, which can be considered as "pattern level" interpretation. By doing so, we can understand the behaviors of black box models by inspecting variable influence not only on the instance level interpretation but also covering the pattern level interpretation.

In summary, the overall goal of this thesis is to develop a multifaceted interpretation framework to explain the inner behaviors of black box models, which should be furnished with various model interpretation methods.

1.3 Thesis Structure

The remainder of this thesis is structured as follows.

Section 2 focuses on previous work on related fields, such as Model Interpretation methods and Subgroup Discovery field. In particular, it is dedicated to review the existing global interpretation methods and local interpretation methods. In addition, the fundamentals of subgroup discovery are discussed as well such as the selection of interestingness measure.

Section 3 is concerned with the theoretical knowledge of local interpretation methods to explain black box models. It begins with simple approaches on specific scenarios, for example, to inspect the influence of binary feature using the binary flip approach. Then the methods become more general which can be applied to any type of features, like Shapley values. Finally, more attention is laid on the novel technique which combines the approach of model agnostic local interpretation and subgroup discovery.

Section 4 presents the detailed description of datasets that are collected and the experiment set up. In experiments, the comparison of several local interpretation methods is covered. Additionally, we demonstrate case studies on specific datasets.

Finally, Section 5 concludes the work with a summary of results and ideas on future work.

2 Background

2.1 Model interpretability

Considerable research efforts have been devoted to interpretable machine learning area with the pressing need to understand the behaviors of black box models. In other words, people would like to ascertain why a black box model makes such predictions. And the extent to explain the model behavior or its predictions in a human-understandable way is termed as interpretability [1]. In this thesis, we will use explainability or comprehensibility as its interchangeable term.

Model explainability can be roughly categorized as two types: intrinsic interpretability and post-hoc interpretability [8]. Intrinsic interpretability refers to models that are inherently interpretable, meaning that its predictions could be explained by model structures and model parameters. On the contrary to that, post-hoc interpretability is achieved by constructing a new model to provide explanations for the black box model. Particularly, post-hoc interpretability is mainly considered in the current context. Aside from the cognitive definition, it has to be noticed that there is no wide-spread mathematical formulae to define or measure the model interpretability. And the assumption that smaller models are more comprehensible than large models concerning the model size is problematic as pointed out by Freitas [9]. More specifically, we might argue that the model complexity is a determining factor to address the measurement of model interpretability. Nonetheless, how to assess the model complexity and how to link these two concepts are beyond our scope.

Basically, models maintaining intrinsic interpretability are interpretable, which are known as interpretable models, including linear models or decision tree models. And it is not surprising that we can easily interpret model predictions through its parameters. For example, we train a logistic regression model to predict the house price. Evidently, we can decompose the house price prediction into the attributions of each feature, weighted by the coefficients. In this regard, an explanation for this prediction could be inferred from the feature impacts. In contrast, black box models are usually have low comprehensibility, with complex model structures and tremendous parameters. For instance, in the booming field of computer vision, practitioners prefer to apply deep neural networks to achieve sufficient performance with complex neural architectures, training procedures, regularization methods, and hyperparameters. Consequently, it is hardly possible for engineers to interpret the result.

Due to the fact the models with intrinsic interpretability are normally interpretable, we hence devote our efforts to the post-hoc interpretability on black box models. And it can be further classified as global interpretability and local interpretability [10]. Correspondingly, global interpretation methods and local interpretation methods are introduced as follows.

2.2 Model interpretation methods

The main difference between global interpretability and local interpretability lies in the view of the dataset to be investigated, as displayed in Fig 1. The former highlights the impact of input variables based on the entire dataset, leading to an overall understanding of features. And the latter implies the justification for a specific decision, targeting at the instance level interpretation. There is a numerous number of papers that have imbued explainability in their methodology, and most techniques could be grouped into global interpretation methods and local interpretation methods, respectively.

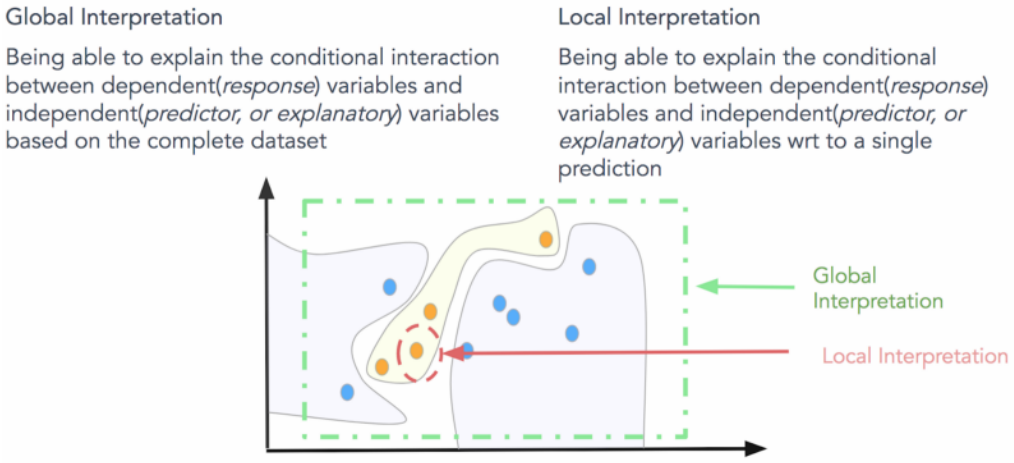

Figure 1: Global interpretation view vs. Local interpretation view. Global interpretation explains the feature impact on a global view, based on the whole dataset. Local explanation focuses on the justification for a single instance by inspecting features.

2.2.1 Global interpretation methods

The global interpretation methods concentrate on the global view of the input variables, more specifically, they identify the most significant features that can largely affect model predictions of the entire dataset. Friedman proposed that the Partial Dependence Plot (PDP) was a global interpretation method which showed the marginal effect of a feature on the model predictions[11]. This method made clear the relationship between the selected feature and the predictions by adapting the values of the selected feature, and to characterize the feature impact on model predictions. Typically, some simple relationships such as linear or monotonic relation could be inferred from the plot directly.

Another popular approach is called feature importance. There are many methods for assessment of feature importance. The default feature importance mechanism was proposed and implemented by the inventor of the RandomForest algorithm, which was to add up the Gini decreases for each variable over all trees in the forest and got the average. However, Strobl et al. had demonstrated that this method was biased and was not reliable in scenarios when the selected variable was biased in terms

of the scale of measurement[12]. Later, an improved strategy called permutation feature importance was described by Fisher et al. [13]. In his approach, the feature importance was estimated by the drop of prediction accuracy of the model after permuting the selected feature. A feature is regarded as "important" if prediction accuracy drops extensively after shuffling feature values as the model depends on the feature for the prediction. Conversely, a feature is "unimportant" if the accuracy is slightly dropped, which means the feature is hardly relied on for the model.

An alternative to permutation feature importance was SHAP feature importance, based on the magnitude of feature contribution using shapley values [8]. To elucidate the idea, we could assume that the model prediction of an individual instance could be decomposed into feature attributions, and each attribution was estimated by the shapley value. Each feature had a corresponding shapley value for each instance. Thus, over the entire dataset, the SHAP feature importance was indicated by the mean absolute shapley values.

2.2.2 Local Interpretation methods

Local interpretation methods aim at the instance level explanation which means each instance should be supplied with an explanation identifying the cause to the prediction. Following this idea, it leads us to the local surrogate methods, which are able to explain individual predictions of any black box models faithfully. As a concrete implementation of local surrogate models, Local interpretable model-agnostic explanations (LIME) was initially proposed by Ribeiro et al. [14]. The general idea was to train an interpretable model to approximate the predictions of the underlying black box model. Since the fitted model was interpretable, we hence could use this explanation model to give detailed explanations.

Another possible approach was to calculate the individual contribution of each feature in an instance to compose the final prediction as described in paper [15]. Inspired by this idea and the theoretical knowledge from the coalitional game theory, shapley value approach was highlighted to explain instance-level predictions with contributions of each feature values [16]. Basically, each feature was assigned an importance score for a particular prediction, and the explanation could be derived from feature importance to some extent.

However, by exploiting the shapley value approach, it was noticed that only a list of shapley values corresponding to each feature was generated to form an explanation for each model prediction, rather than an explanation model such as LIME, which failed to make judgments about the connections between input changes and prediction changes. To address those problems, Lundberg and Lee [17] proposed a unified framework for explaining predictions, which was based on the shapley value, and it was named SHAP(SHapley Additive exPlanations). In this unified framework, there was an novel approach called Kernel SHAP, which was the combination of linear LIME and shapley values. In this way, the intuitive connections between these two methods made this approach more promising. Besides, potential techniques to solve the computational performance problem in KernelSHAP was brought up as

well. Tree SHAP, one of the variants in SHAP framework, was exhibited to deal with the computational complexity problem particularly for tree-based black box models. It implemented fast and efficient algorithms to calculate shapley values in comparison to Kernel SHAP. In addition, Deep SHAP was designed to improve computational efficiency when deep neural network was applied.

2.3 Review of subgroup discovery technique

2.3.1 Definition

Recent developments of the research filed in knowledge discovery in databases have attracted much attention, where numerous methods are proposed to extract local patterns from large volumes of data [18]. Apart from the methods for mining local patterns such as discriminative patterns [19] and emerging patterns [20], subgroup discovery (also called pattern mining) is established as a supervised and descriptive data mining technique. As defined in [21], in the subgroup discovery task, assuming we have a population of individuals and the corresponding property of interest, it aims to discover subgroups that are statistically "most interesting". To put it another way, the interesting subgroups have the most unusual distributional characteristics with respect to certain property of interest given by the target variable [22].

In a formal definition, the fundamental concepts of subgroup discovery task could be summarized by a quadruple (D, Σ, T, Q) [23]. In the quadruple, D represents the dataset, which is formed by a group of instances. Σ means the search space, consisting of a set of selection expressions, and the search space covers all the patterns that are going to be traversed through. Take an example, one of the selectors could look like: "sex=Male AND age>30". T implies the target concepts being exploited in the pattern mining task. Commonly, a single target concept, e.g. binary or numeric, is applied to the mining task. Nevertheless, multi-target concepts are also allowed given the existence of exceptional model mining framework [24]. Concerning the quality measure criteria, symbolized as Q, it is specified depending on the target concept.

2.3.2 Quality measure

To gain more insight, we present some quality measure criteria in this part. Since considerable research efforts have been devoted to study the binary target concept, the quality measure for binary target is well-investigated. One variant of the quality measure for binary target could be easily estimated by the parameters contained in a contingency table, which describes the distribution of positive/negative instances for the observed pattern and its complement subgroup, respectively. According to an investigation by Kloesgen et al. [25], they proposed a prevalent family of quality measure, relating to the size of the subgroup and the difference between the target share in the subgroup and the target share in the general population.

Correspondingly, several approaches to measure the quality of numeric attributes had been proposed, and a list of interestingness measures for numeric target concepts could be found in paper [26]. Since a numeric attribute has certain characteristics, such as mean value or median value, therefore, the quality measure for a numeric target could be formalized by slightly adapting the quality function which is designed for binary targets. To be specific, chances were that the share of target in the subgroup and in the entire population could be replaced by the characteristic of the target. Generally, there were five categories of interestingness measure for numeric target, concluded by Lemmerich [23], which were mean-based measures, median-based measures, variance-based measures, distribution-based measures, and rank-based measures. Furthermore, as for a multi-target concept, the quality function had been described in a number of studies. And a general framework for multi-target quality functions was the exceptional model mining framework reported by Leman et al. [24], proposing a variety of model classes, which contained the correlation model, the regression model, and the classification model class.

2.3.3 Subgroup discovery algorithms

From previous part, four indispensable components were mentioned to define the problem of subgroup discovery. And in the following, algorithms to efficiently execute the subgroup discovery task is going to be considered.

Unlike the choice of the quality measure which is mainly determined by a target concept in the subgroup discovery task, the mining algorithms are almost equivalent. And for a specific algorithm, three algorithmic components should be verified, which are enumeration strategy, data structure, and pruning strategy. Various enumeration strategies could be used, e.g. exhaustive methods, seeking to acquire the optimal subgroup by traversing through the whole search space. In contrast, heuristic approaches, normally a beam search strategy, was often used for subgroup discovery due to its efficiency, which aimed to find interesting patterns but not necessarily the optimal patterns in a short time [27]. From data structure perspective, data was normally stored in a horizontal layout, e.g. tabular-formatted database. Instead, vertical data representations could also be used, which was covered in paper [28]. Besides, referring to the wide-spread FP-Growth algorithms, FP-tree structure was also applicable to data [29]. Furthermore, considering the efficiency of algorithms, the pruning strategies was of critical importance. To determine the upper-quality bounds and safely prune parts of the search space, optimistic estimates could be explored as initially stated by Wrobel et al. [30]. In addition, to shrink the search space of subgroup discovery task, minimal support pruning strategy was useful by exploiting anti-monotone constraints.

3 Approach

In this chapter, the details of approaches will be discussed. It starts with an overview of local interpretation methods and several variants of them are introduced respectively. Firstly, we present the binary feature flip idea which aims to characterize the impact of binary features by flipping the feature values. After that, we are interested in the effect of numeric features in the model by inspecting the outcome change of the model when the numeric feature values are perturbed to generate noises. Despite the "variable-specific" methods, we also focus on local interpretable model-agnostic explanations (LIME) which is able to explain individual predictions for any types of features and models. However, no theory can support why LIME can fit linear behavior locally on black box models. Therefore, we continue exploring Shapley value, which is a reasonable explanation method with well-founded theory. In addition, the appealing approach assigns a contribution score for each feature value to smooth the path of interpreting the final prediction of individual instances by calculating the Shapley value.

Then the following section describes the novel technique which combines the local interpretation methods and pattern mining technique. Since the target concept during subgroup discovery in our situation is either prediction change or feature influence score, therefore, the focus shall attribute to numeric target. Later, the standard approaches to measure the interestingness of subgroups are discussed. Furthermore, methods to avoid redundancy in subgroups are explored.

3.1 Structure of interpretation framework

As you probably have noticed that the adoption of complex machine learning models which have high performance is growing rapidly. Driven by the boosting awareness of machine learning field, the urgent need to interpret those complicated black box models is occurred. Besides, external pressures are withstood by the enforcement of regulations like GDPR in EU. Given the situation, the research about interpretable machine learning are quite active.

Generally, many existing approaches for interpreting black box models are based on the model prediction, i.e., generating an explanation for the model prediction by inspecting the variable influence from the input, which is operated on the instance level. Concluded by Alvarez-Melis [4], those methods can be roughly categorized as salience-based and perturbation-based approaches. The former method category is also known as gradient-based attribution methods, computing the partial derivatives of the output with respect the each input feature, e.g. Integrated Gradients [5][6]. In contrast, perturbation-based approaches first generate a bunch of neighborhood data points surrounding the instance to be explained, then calculate the contribution of each input features towards the output by fitting a local interpretable model, e.g. LIME [7].

With those theoretical methods, a list of machine learning interpretability framework are surfaced. For instance, DeepExplain is a unified framework of perturbation and

gradient-based attribution methods for deep neural networks interpretability which can be found in [31], referring to the method presented by Marco[32]. Another framework is LIME, which supports explaining the predictions of any machine learning classifiers, founding on the perturbation-based approach. And the framework is available in [33]. In addition, a promising framework called SHAP could connect the game theory with local explanations to provide understandable interpretations for any black box models, which is accessible and open sourced on [34].

After seeing these frameworks, it could be easily observed there are huge drawbacks for each single framework. DeepExplain is mainly designed for image classifier that is using deep neural network. Even though LIME is applicable to tabular data, text data, and image data, the framework support for complicated models such as deep neural network is not well implemented. And the the exact evaluation of Shapley values is prohibitively expensive in SHAP framework, which cannot be used as an online algorithm. What is worse, even non of those frameworks include the global interpretation view on the black box models, which is handy in many situations.

In light of those flaws, we aim to construct a new interpretation framework. From a large point of view, the global interpretation methods and local interpretation methods are included. For a global interpretation, feature importance ranking is supported by the permutation feature importance method. On the other hand, the existing local interpretation methods like LIME and SHAP are incorporated into the new framework. Besides, two very simple model-agnostic approaches, named as binary feature value flip and numeric feature value perturbation respectively, are implemented as well. Nevertheless, the most important contribution is that we manage to derive a mid-level interpretation of black models between global interpretation view and local interpretation view by combining the local interpretation methods and subgroup discovery technique. In this way, we are supposed to discover interesting patterns in dataset where the inspected variable impose large impact.

In the following, two large components in the framework will be introduced separately, which are local interpretation methods and subgroup discovery technique.

3.2 Local interpretation methods

In comparison to Global interpretable methods which are dedicated to explain the global model output by comprehending the entire datasets, it is more interesting to examine the model prediction for an individual instance. Besides, it could be observed that the global interpretation methods are less sensitive to noises if we make some perturbations on feature values, however, it could lead to tremendous changes in the prediction for an instance. Therefore, the local explanations shall preserve high accuracy than global explanations. In the following, few local interpretation methods will be covered in detail.

3.2.1 Binary feature value flip

Binary feature implies that the feature only contains two unique values. In another word, if it is encoded as discrete numeric number, the feature value should be either 1 or 0. Thus, to flip binary feature value means to convert from 1 to 0 or the other way around. In practice, we could also use the XOR operation to map from 1 to 0. For instance, gender is regarded as a binary feature which only holds value "male" and "female".

As mentioned previously, the assumption is that we hold the dataset and the corresponding model trained on that dataset. Initially, we could obtain the prediction from the model for a specific instance. Then, a binary value is flipped on a chosen feature and afterwards a new prediction is generated by applying the model to the modified instance. Therefore, as a simple measurement, the effect of this binary feature could be estimated by the difference between two outputs.

In practice, there are two variants to assess the variable influence. One way is to calculate the absolute difference of two predictions, and in this way we could ignore the bias of this binary feature on the original dataset. Literally to say, the binary feature is more influential when the difference becomes larger. In contrast, we could compute the difference for a defined direction, for example, we just care about the effect of gender changing from male to female. In this case, not only the magnitude of the effect is obtained, but also the positive or negative sign towards the prediction.

3.2.2 Numeric feature value perturbation

As the name suggests, this technique is applicable to features whose type is numeric. The idea is that we could apply binary operations to the input values to produce new values, which serves as injecting noises into the original dataset. In particular, only addition and subtraction are considered in this situation. For example, an instance includes a numeric feature called "age" and we could perturb this feature value by increasing or decreasing by a certain value to obtain the modified value.

The procedure of measuring the effect of a chosen input feature is similar to that in binary feature value flip approach. For classification or regression tasks, we could make predictions with the existing model on the instance we desire to explain. Afterwards, a new prediction is made on the adapted instance which is produced through perturbation on the selected numeric feature. And the impact of this numeric feature could be approximately evaluated by the absolute difference of two output predictions, which indicates that this particular feature plays an important role in this instance, causing unstable predictions. Roughly to say, larger prediction differences might imply the feature has a stronger effect on the corresponding instance.

3.2.3 LIME: Local Surrogate

Various criteria can be used to classify types for machine learning interpretability. Intrinsic interpretability, for example, is one type of the interpretability methods, which refers to models that are intrinsic interpretable owing to their simple structures, such as linear models or decision trees. In contrast, post hoc interpretability is meant to analyze the model interpretability after model training. As introduced earlier, permutation feature importance is a post hoc interpretation method.

In this thesis, we would like to focus on post hoc interpretability, which indicates to explain model decisions after the model has been trained. In particular, model agnostic interpretation methods, which extracts post hoc explanations by treating the original model as a black box, is highly valued. The model agnostic interpretation method is pretty flexible in terms of models, and it can work with any type of machine learning models, which provides a great advantage over model-specific methods [7]. The principle behind is to learn an interpretable model on the decisions of the black box model and in return apply the interpretable model to those predictions that are expected to explain.

Following this idea, it leads us to the local surrogate methods, which are able to explain individual predictions of any black box models in a faithful way. As a concrete implementation of local surrogate models, Local interpretable model-agnostic explanations (LIME) was initially proposed in paper [14].

The key point behind LIME is pretty straightforward. It is intended to explain individual explanations by fitting a simple interpretable model to locally approximate the underlying black box model. The typical choice of the interpretable model could be regularized linear models like Lasso or decision trees. To elaborate more intuition for LIME, the toy example is shown in 2. This is a binary classification task and the regions colored with blue or pink are regarded as two distinct decisions. Evidently, this decision function can not be easily interpreted by a linear model. As a clarification, we are interested in the individual instance explanation, which is marked with a bold red cross. To fit a local interpretable model, some artificial points are created by perturbing the original data point. The learned local model, marked by the dashed line, could in principle provide a faithful explanation for the target instance.

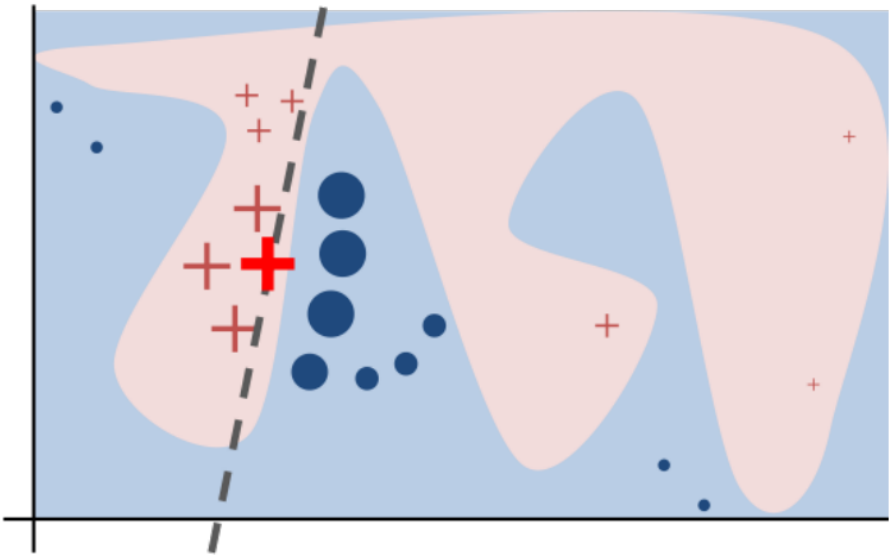

Figure 2: Binary classification task for a black box model. Decision regions are colored with a blue or pink background. Instance to be explained are marked in a bold red cross. Artificial points, marked as crosses and circles, are created by perturbing the instance of interest, whose size are weighted by the proximity to the instance. The dashed line expresses the fitted local interpretable model which could give faithful explanations.

Apart from the intuition, we could argue for the faithfulness from a mathematical perspective and the constraint of LIME could be represented as equation 1.

$$\xi = \arg\min_{g \in G} L\left(f, g, \pi_x\right) + \Omega(g) \tag{1}$$

As formally defined in the equation, f is the black box model, g is the local explanation model needs to be figured out, and G is a group of interpretable models, which includes linear models, decision trees, or falling rule lists [35]. As depicted in the figure, the weight is measured by the proximity of instance of interest to the surrounding artificial instances, which is defined as $\pi_x(z)$. And the complexity of explanation model g is described as $\Omega(g)$. For example, the complexity could be estimated by the depth of trees for decision trees models or by constraining the maximum number of features in linear models. Thus, as seen from the formula, in order to obtain the local explanation model for instance x, the loss L (e.g. mean squared error) should be minimized while maintaining the complexity as low as possible.

In practice, the general procedure to train an explanation model is described as follows: First, select an individual instance that we desire to explain for its black box prediction. Then, generate artificial data points by perturbing the selected sample and make predictions for these new instances using the original model. Afterwards, calculate the weights for new instances according to their proximity to the instance being explained. Next, fit a weighted, interpretable model on the obtained dataset. Finally, interpret the instance prediction by utilizing the trained local interpretable model.

After a literature review, it is found that LIME is one of the few methods that work for tabular data, text and images, which is a very promising approach. The python implementation is currently available in [33], which is still in active development and needs further exploring.

3.2.4 Shapley values

As we have seen, numerous approaches have been recently proposed to explain predictions for individual instances of black box models. As stated in [15], the presented approach is relied on the decomposition of a prediction for a single instance on individual contributions of each attribute, and the contribution for each feature value is measured as the difference between the output value and the average output over all perturbations of the corresponding feature. Nevertheless, this approach fails to work if the features are conditionally dependent.

Inspired by the coalitional game theory which instructs us to fairly distribute the "payout" among the "players", a general method for explaining black box models by taking into account interactions between features can be found in [16], whose fundamental concepts are borrowed to explain instance-level predictions with contributions of each feature values. Corresponding to the known concept in coalitional game theory, the contributions of individual feature values are called Shapley Value.

Despite from the abstract concept, an illustration taken from [8] might help us intuitively understand the Shapley value. Imagine there is a room and all feature values of an individual instance enter the room in a random order. All feature values, seen as players, need to collaborate with each other to participate the game, where each player contributes to receive the final prediction. And each order of feature values represents a coalition. Consequently, the Shapley value of a feature value corresponds to a difference in the value of a coalition when the feature is added to it. In other words, the Shapley value is the average marginal contribution of a feature value across all possible coalitions.

Then, let us have a detailed look at the formal definition of Shapley value as expressed in equation 2, where S is the subset of the features in an individual instance, p is the number of features, and x is the vector of feature values of the instance to be interpreted. As for characteristic function val, it describes the contribution of feature j in each coalition.

$$\phi_j(val) = \sum_{S \subseteq \{x_1,\ldots,x_p\} \setminus \{x_j\}} \frac{|S|!(p-|S|-1)!}{p!} \left(\text{val}\left(S \cup \{x_j\}\right) - \text{val}(S)\right) \tag{2}$$

Referred to [36], the Shapley value can provide the unique solution that adheres to he desirable properties, which are Efficiency, Symmetry, Dummy, and Additivity.

Efficiency: denoted as 3, which requires that the sum of feature contributions must equal to the difference of the final prediction and the average prediction over all coalitions.

$$\sum_{j=1}^{p} \phi_j = \hat{f}(x) - E_X(\hat{f}(X)) \tag{3}$$

Symmetry: The contributions of two feature values j and k are the same, which means equation 4 should be satisfied.

$$\begin{aligned} if\ val\,(S \cup \{x_j\}) = val\,(S \cup \{x_k\}) \\ then\ \phi_j = \phi_k \end{aligned} \tag{4}$$

Dummy: The contribution of feature j is 0 if it does not change the predictions when it joins into any coalitions. This properties can be demonstrated in equation 5.

$$\begin{aligned} if\ val\,(S \cup \{x_j\}) = val\,(S) \\ then\ \phi_j = 0 \end{aligned} \tag{5}$$

Additivity: For any pair of games v, w, the combined payouts should equal to the sum of two individual payouts, as shown in equation 6. For example, if we trained a random forest and the additivity axiom guarantees that we can calculate the Shapley value for each tree respectively then average them to obtain the final Shapley value.

$$\begin{aligned} \phi_j(v + w) = \phi_j(v) + \phi_j(w) \\ where\ (v + w)(S) = v(S) + w(S) \end{aligned} \tag{6}$$

3.3 Kernel SHAP

Though classical Shapley value leads to a potentially promising result, this approach is too computationally expensive owing to computations for the exponential number of possible coalitions. Feasibly, approximation algorithms could be used to reduce the computational complexity, nevertheless, it inevitably will increase the variance for the calculation of Shapley value. What is worse, the explanation for the prediction of a model is just a simple value, rather than an explanation model like LIME, which fails to make judgments about the connections between input change and prediction change. To address those problems, Lundberg and Lee [17] proposed a unified framework for explaining predictions, which is based on the Shapley value, and they named it SHAP(SHapley Additive exPlanations). This novel approach unifies existing explanation methods and brings more clarity to the methods space. They introduced the explanation model by treating the explanation of an individual prediction as a model. Of course, the unique solution is guaranteed with the game theory. In addition, it provides a more human-understandable and intuitive explanation by user studies as they claimed.

In this case, SHAP values are introduced as a novel measure of feature contribution. Similar to classical Shapley value estimation methods, SHAP values provide the unique additive feature importance measure if the flowing properties are satisfied, which are Local accuracy, Missingness, and Consistency [17]. From another perspective, SHAP method transforms the Shapley value approach into an optimization problem by using kernel function to measure proximity of instances. Within this domain, the novel approximation model agnostic method is called kernel SHAP, which is a combination of LIME and Shapley value. In order to use linear explanation model to locally approximate predictions, we should minimize the following objective function 1.

It is intended to obtain the unique solution of equation 1, which should also be in line with those three properties, the Shapley kernel is defined as [17]:

$$\begin{aligned} \Omega(g) &= 0 \\ \pi_{x'}\left(z'\right) &= \frac{(M-1)}{(M \text{ choose } |z'|)\,|z'|\,(M-|z'|)} \\ L\left(f, g, \pi_{x'}\right) &= \sum_{z' \in Z} \left[f\left(h_x\left(z'\right)\right) - g\left(z'\right)\right]^2 \pi_{x'}\left(z'\right) \end{aligned} \tag{7}$$

where $|z'|$ is the number of non-zero elements in z'

The SHAP framework recently is in active development and is accessible at [34]. Since it is seemed to be a very optimistic approach and we are quite interested in this novel model explanation method, therefore, further experiments will be conducted.

3.4 Pattern mining with Local interpretation methods

It appears from the aforementioned sections that a bunch of local interpretation methods which could be exploited to explain an individual black box model prediction were incorporated into our interpretation framework. Attempts were made to investigated those local interpretation methods. In principle, it was assumed that we could obtain a local explanation model for each model prediction independent from the underlying black box model. To put it another words, the instance-level explanation for a chosen prediction shall remain consistent even though the underlying black box model was modified. Nevertheless, situations could happen that two disparate explanations were provided for the same instance prediction when two different underlying black box models were applied. This unexpected results may be attributed to the excessive interpretation of the selected instance, which was merely one sample instance rather than a representative of all instances. Moreover, insights extracted from an individual instance might be too specific to train a well-fitted explanation model, causing unstable and unreliable explanations.

Recall from previous part, global interpretation methods and local interpretation methods could provide explanations from a global view point and a local view point on the dataset, respectively. Therefore, to overcome the drawbacks from the global view or local view, we would naturally consider to interpret the black box model

from somewhere between the global view and the local view. Inspiration from the idea that black box models could detect hidden patterns in a dataset such that they could exhibit a good performance when executing classification tasks, it came to our mind pattern mining was exactly the technique could be utilized to discover hidden patterns in a dataset, which coincided with each other somehow. Therefore, a novel method by combining the local interpretation methods and pattern mining technique was proposed in this thesis. In this regard, it was presumed that the novel approach could provide a "pattern level" view point on the dataset.

To be frankly, it brings more clarity if we discuss these two individual components separately, which also correspond to two techniques. Since we have already covered the discussion about local interpretation methods, it is intended to elaborate the subgroup discovery technique in the following section.

3.4.1 Overview of subgroup discovery

Note that the terminology pattern mining, whose interchangeable name is subgroup discovery, refers to a data mining technique which pursues to find subgroups of data instances that exhibit interesting characteristics with respect to a predefined target variable [21]. Moreover, subgroup discovery is a descriptive technique, which describes enough details such that the results are understandable by human experts.

In a more formal definition, four elements could be considered the fundamental components to compose the subgroup discovery task, which is defined by a quadruple (D, Σ, T, Q). These elements are illustrated as below [37] [23]:

- D is a dataset and is formed by a set of instances, and each instance consists of a set of attributes
- Σ constrains the search space, which is made up of subgroup descriptions (patterns). And patterns consist of a set of selection expressions, also known as selectors.
- T represents the target variable for the discovery task. Various types of target concept could be identified, including binary target, numeric target, or complex target.
- Q defines the quality measure criteria. Different quality measure criteria are specified for different types of target concept.

In principle, the dataset could be any kind of data type, nevertheless, we will primarily concentrate on tabular data, textual data, and sequence data. Actually, the detailed description about the datasets that are used in this thesis will be discussed in the next section, and thereby it will not be deeply investigated here. As for the search space, commonly it is accepted as conjunctive combinations of selectors for the reason that such subgroup descriptions are interpretable by practitioners. As an example, a pattern could be formatted as: $P = sel_1 \wedge sel_2 \wedge ... \wedge sel_d$, where all selection expressions are evaluated to be true. Loosely speaking, the full search space is

exponential to the number of input features of the dataset, which can significantly affect the pattern mining efficiency. By taking the size of the search space into account, beam search strategy is adopted to shrink the search space in order to speed up the subgroup discovery task and more detailed information will be illustrated later.

The choice of the target concept is normally task driven and closely related to the dataset. Using a binary variable as the target of subgroup discovery is a more simple and general situation. Since the binary variable only contains two values (True or False), it is aimed to identify interesting subgroups for each of the possible value. Basically, the idea is to discover patterns whose target share is either remarkably high or remarkably low. However, in this thesis domain, it is desired the discover subgroups which reveals a significant effect of the inspected variable, meaning that the influence scores of the selected attribute are considered as the target concept, which belong to the numeric data type. Generally, pattern mining for numeric target is more complicated because the attribute values could be handled by a numerous approaches such as numeric target discretization in a predefined number of intervals, or dividing the numeric domain into two ranges with respect to the average. And frequently mentioned discretization methods includes equal-width discretization, equal-frequency discretization, and etc. An overview of discretization methods for a numeric attribute was reported by Garcia et al. [38]. In this thesis, it is more inclined to apply the equal-frequency discretization method.

Without any doubt, it is critical to choose the quality measure carefully, since the results of subgroup discovery are mostly controlled by the quality measure criteria. In light of the fact that the interestingness measure plays a decisive role in the subgroup discovery task, thus, a comprehensive discussion about quality measure for numeric target will be presented in the following subsection.

3.5 Interestingness measure for numeric target

As was said before, the interestingness measure for numeric target becomes more complex to investigate than situations where binary values was chosen as the target. Nevertheless, a list of interestingness measures for numeric target was reviewed by Pieters et al. [26]. As could be summarized, those interestingness measures were heavily relied on the basis of the statistical distribution of numeric values, such as mean value, median value, or variance. And the general idea behind was to design the interestingness measure for numeric target with respect to those predefined data statistics. More specific, interesting subgroups would be discovered if the computed data characteristic in the subgroup was significantly deviating from the value calculated in the entire population. Referred to paper [23], five categories of interestingness measure for numeric target were outlined, which included mean-based measures, median-based measures, variance-based measures, distribution-based measures, and rank-based measures. Since the mean-based measure was widely accepted and applied in many applications, therefore, it was selected as the primary quality measure for numeric target in later experiments.

In point of fact, within this mean-based measure family, several concrete interestingness measures could be further explored, distinguishing by the evaluation functions. Take an example, one simple evaluation function could be *Average function*, which calculated the difference between the mean value in the subgroup and the mean value in the entire dataset, denoted as: $q_{mean} = \mu_p - \mu_\emptyset$. However, subgroup size was not considered in the former measure, which might be too fragile in certain circumstances. Actually, it was often observed from literature that *Generic mean function* was the most prevalent mean-based interestingness measure due to its simplicity to be interpreted. And the general formulation is denoted in Eq. 8, where i_P was the size of the subgroup, a was a parameter which weighted the subgroup size and deviations, and $\mu_P, \mu_\emptyset$ represented the average value in the subgroup and the average value in the dataset, respectively. In particular, the choice of parameter a could be selected in an iterative process. For example, a was required increment if the subgroup size was too small to have a significant score, meanwhile, low parameter values for a was preferred with a high deviation of mean target values between the subgroup and the overall dataset. Therefore, after calculating the interestingness score for each subset, those subgroups with significantly higher or lower mean values were considered as interesting and the descriptions of them were our desirable interesting patterns.

$$q^a_{mean}(P) = i^a_P \cdot (\mu_P - \mu_\emptyset), a \in [0, 1] \tag{8}$$

3.6 Algorithms

Apart from the quality measure, the search strategy is critical since the dimension of the search space and time complexity if of great concern. Various strategies could be used, e.g. exhaustive methods, seeking to acquire the optimal subgroup by traversing through the whole search space. In contrast, heuristic approaches, normally a beam search strategy [27], is often used for subgroup discovery due to its efficiency, which aims to find interesting patterns but not necessarily the optimal patterns in a short time. The intuition behind is that it is assumed that the patterns are more likely to be interesting if their generalizations are also interesting. Therefore, the search starts with an empty hypothesis, then it tries to find the best patterns with size k (corresponding to beam width) by evaluating all selectors in the subgroup discovery task. Following that, at each search iteration, the hypotheses contained in the beam are expanded but only the currently best w hypotheses are kept using a hill-climbing greedy search [39].

3.7 Redundancy avoidance

Though patterns could be discovered through the traditional interestingness measure presented above, the results are not ideal, which contains too many redundant patterns. In the quality measure, only the subgroup size and statistics difference between subgroups and entire dataset are considered, which might produce uninter-

esting patterns when ignoring the selector expressions of subgroups. For example, assume that the mean contributions of age for the entire dataset is at $M_{\emptyset} = 0.50$. And the mean value in the subgroup with the expression $age > 40 \cup gender = male$ is $M_{age>40 \cup gender=male} = 0.80$. It seems that the pattern should have a high quality score and is identified as an interesting pattern. However, it is probably not interesting enough if given the information that its generalization has nearly the same value, which means mean value does not deviate significantly from the mean value of its generalizations, e.g. $M_{age>40} = 0.78$.

To avoid that such subgroups are included in the result set, Generalization-aware interestingness measures could be applied to improve the traditional selection criteria for pattern mining by considering the statistics of the subgroup and also to its all generalizations. In [40], Grosskreutz et al proposed to estimate the quality of a pattern P as the minimum of the quality of P with respect to the extension of all its generalizations. Denoted as equation 9, q^{Δ} is the incremental version of q, D is the dataset, P is the subgroup and H includes its all generalizations.

$$q^{\Delta}(DB, P) = \min_{H \subset P} q\left(DB\left[H\right], P\right) \tag{9}$$

Since the mean value of the target is mainly explored, the above equation could be formalized in a simpler way, as shown in 10. By doing so, redundant patterns are avoided and more interesting subgroups are discovered.

$$q^{a}_{\text{mean}}(P) = i^{a}_{P} \cdot \left(\mu_P - \max_{H \subset P} \mu_H\right), a \in [0, 1] \tag{10}$$

3.8 Combination with local interpretation methods

Recall from local interpretation methods, the contribution values for each variable of an individual instance could be obtained by using the explanation model to interpret the black box model. In this case, the contribution score of the chosen variable is our target, which is naturally to be numeric. Now, the next step goes to the traditional pattern mining problem, aiming to discover subgroups of the population that are statistically interesting.

3.9 Decision trees with Local interpretation methods

The Decision Tree is a supervised machine learning method used for classification and regression tasks. It is called "Decision tree" because the structure of each decision tree is a tree-like graph and model is constructed to predict the target by learning simple decision rules inferred from data attributes. Decision tree itself is an interpretable model that can provide human-understandable decisions by exploring the decision rules. To be specific, a decision tree applies a recursive partition technique, which keeps on splitting the data based on the selected attributes. From

another perspective, it is also a predictive rule-based approach, which could be used to mine local patterns through the decision path, where each path is traversed from the root node to a leaf node. In literature, numerous methods to create decision trees have been proposed and the main difference between decision tree induction strategies is in their attribute selection methods [41]. (Besides, it is worth to be mentioned that the decision tree finds the optimal splitting pattern by essentially limiting the number of conditions to one. [42])

4 Experiments

After a detailed introduction to the methodology, in this chapter, we will concentrate on practical experiments. To begin with, datasets that will be utilized in the experiments will be introduced, including the well-known UCI datasets and the artificial dataset. Next, a general overview of the classification or regression models that will be encountered in the experiments is provided. Then, we intend to compare different local interpretation methods on the same dataset and give explanations on the feature that is interested. To interpret the feature explanation for an individual instance and discover the interesting patterns, the subgroup discovery technique is applied. As a comparison, decision tree visualization is exposed to find local patterns. And finally, we will apply the promising SHAP approach to conduct case studies on real datasets.

4.1 Datasets

4.1.1 Artificial dataset

Before exploring the local interpretation methods on real datasets, we would like to justify the concept that interesting subgroups could be recovered from the artificial dataset by inspecting variable influence. Presumably, there were hidden patterns in the synthetic dataset that were useful to provide a reasonable explanation for the predictions. By interpreting the effect of a certain variable, e.g. gender, it was assumed that the interesting pattern could be recovered. The procedure to construct an artificial dataset and conduct experiments will be described as follows.

For simplicity, we constructed the artificial dataset relying on the popular "Adult Income" dataset, but we only extracted partial information, which meant that only the information about age, education-num, sex, hours-per-week and income were included. As assumed, the synthetic data contained some interesting patterns, such as "age $<$ 30". One exemplary case was that when "age $<$ 30", the attribute "gender" had a stronger effect on predictions while in its complementary subgroup, the effect of "gender" was slight. And the task was indeed to discover this pattern by exploring the effect of gender.

For further experiments, one way to fabricate this interesting pattern was to modify the gender effect directly on the corresponding subgroups. For instance, if the condition that "age $<$ 30" was met, we could manually add 3 unit in terms of the scale of measurement on gender effect, and otherwise we could subtract 3 unit. Another idea was to establish two models that behaved differently when considering this condition. It is known that the coefficients in the logistic regression model have straightforward interpretation, indicating the influence level by the input features. Therefore, we could create two distinct models by changing the weights of the features in accordance with the previously defined patterns, which was that when "age $<$ 30", the effect of gender was relatively large. In specific, we could assign larger weights to the model that was applied to the pre-described subgroup to maintain

larger gender effect, while decreasing feature weights on the model that was applied to its complementary subgroup.

In this paper, we would like to adopt the latter method to make up the synthetic dataset and build the models.

4.1.2 UCI datasets

Apart from the synthetic dataset, we will mostly consider datasets that could be found in UCI Machine Learning Repository [43][44]. Ideally, we would like to choose datasets that cover various domains, including social, financial and life science areas. Therefore, for classification tasks, concerning the popularity and quality of datasets, we decided to adopt the "Adult Income", "German Credit", and "Breast Cancer Wisconsin" datasets. In Adult Income dataset, there are 14 descriptive features and more than 40 thousand instances, which were extracted from the US Census database. And the task was to predict whether a person earned more than 50K a year or not. As for the German Credit dataset, it was determined to figure out whether a person had good or bad credit risks relying on the 20 descriptive attributes for each person. It is worth mentioning that these two datasets contain multivariate data types, consisting of categorical features and numerical features. In that regard, data preprocessing needs to be considered in addition. Another Breast Cancer dataset is composed of 32 features and all of them are numerical features except for the predicted label which tells whether the diagnosis of cancer is malignant or benign. Those features of an individual instance are extracted from an image of a breast mass, which describes the characteristics of the cell nuclei in the image.

For regression tasks, we specifically choose the "Bike Sharing" and "Boston Housing" datasets. In Bike sharing dataset, the task is to predict the count of total rental bikes within a specific time frame. It is made up of 17389 entries and each with 16 distinct features. Regarding the Boston housing dataset, it is derived from US census service concerning housing price in the area of Boston MA. 505 records can be found in the dataset and each record contains 14 numerical features.

In summary, a general overview of real-world datasets that will be used in experiments is concluded in Table 1

Table 1: Datasets used in experiments

Datasets	Usage	#Instances	#Features
Adult Income	Classification	48842	14
German Credit	Classification	1000	20
Breast Cancer	Classification	569	32
Bike Sharing	Regression	17389	16
Boston Housing	regression	505	14

4.2 Experiments setup

4.2.1 Machine learning models

The first machine learning algorithm that will be used in experiments is Random Forests, which are an ensemble method for classification or regression tasks by creating multiple decision trees at the training time. In this algorithm, it uses bagging and feature randomness when constructing each individual tree. And the final prediction is decided by the voting among a large number of independent trees [45]. The key concept behind this algorithm is that a large number of relatively uncorrelated trees operating as a committee will outperform any of the individual constituent models. Typically, random forests are treated as a black box model since it is nigh infeasible to gain a full understanding of the decision process by examining each tree. Commonly, the implementation of this algorithm in the scikit-learn library is adopted.

Another black box model is Gradient Boosting Trees, which also construct an ensemble of decision trees to perform classification or regression tasks, where each decision tree is a weak prediction model. However, unlike Random Forests algorithm that fully grown decision trees are created, in Gradient Boosting Trees algorithm, each tree is a shallow tree, sometimes even as small as decision stumps (trees with two leaves). The main idea behind is to add new decision trees to the ensemble sequentially. At each iteration of the training process, those data instances with high prediction errors are emphasized by the next decision tree in order to correct the errors. And the final prediction is determined by the weighted average for each decision tree, where the weight depends on a performance of the corresponding tree [46].

There are a rich variety of libraries that implement the gradient boosting trees algorithms. In this thesis, two efficient and scalable implementations are mainly adopted, one is called XGBoost [47] and the other is LightGBM [48]. XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible and portable and it was a part of winning solutions of multiple machine learning competitions [49]. The library also works natively with scipy sparse data format and can convert it to an internal data format, called DMatrix, which speeds up the training process. Comparatively, LIghtGBM also implements fast, distributed, high performance gradient boosting algorithms. As claimed, it can outperform existing frameworks on both efficiency and accuracy with significantly lower memory consumption [50].

Recently, Rudin pointed out that people have a blind belief in the myth of the accuracy-interpretability trade-off, meaning that there is a widespread acceptance that more complex models have higher performance [51]. And that is one of the reasons why neural network is applied to many fields and is believed to provide the state-of-the-art performance. Since we aim to interpret any kind of black box models in this framework, thereby we believe it is worth to investigate neural networks. There are many types of architectures for neural networks, but for simplicity, full-connected layers neural network is chosen for the experiment.

On the other hand, we design the framework to be compatible with text classification. In text classification task, we usually process the text as sequence of words. However, it is noticed that the fully-connected layers neural network fails to process sequential data efficiently. Therefore, we determine to use an architecture called Long short-term memory network (LSTM), which is an variation based on recurrent neural networks.

4.3 Recover patterns on artificial dataset

As clarified earlier, the artificial dataset was constructed based on the Adult dataset with a hidden pattern indicating that the gender had a large impact on the prediction when "age $<$ 30". Therefore, the aim was trying to verify whether this interesting subgroup could be recovered by pattern mining technique.

Firstly, to measure the gender effect, we could simply use the binary flip approach described in the previous chapter. By flipping the gender value, i.e. transform from "male" to "female" or the other way around, the prediction change denoted as probability was calculated and roughly it was regarded as the effect of gender. Then, treating the effect of gender as the target concept, the subgroup discovery technique was applied to the artificial dataset to discover interesting subgroups. It could be observed that these interesting subgroups include the subgroup that was artificially generated in the dataset. The detailed results were left to the next chapter. In conclusion, it could be proved that the subgroup discovery technique could indeed provide us patterns of explanations that facilitate us to understand the predictions.

4.3.1 Comparison of different local interpretation methods

In previous chapter, we have already introduced several local interpretation methods, and in this subsection, we would like to have a detailed comparison of those methods. In this experiment, breast cancer dataset is selected. First of all, concerning the influence by the interactions between features, related features should be dropped through feature processing. Generally, the pearson correlation coefficient between features are explored and some highly correlated features are excluded. Since we intend to inspect the variable influence, it is better to have an overview about the feature importance. Thereby, it is aimed to explore the impact of the most important feature by various local interpretation methods.

In this case, since we know that the dataset contains only the numeric features, we would first use the numeric perturbation method to estimate the impact of the most important feature for a specific instance. Then LIME could be utilized to fit a ridge regression model for the selected instance, and the coefficients represent the feature weight, which provides implications for the explanation. In contrast, Kernel SHAP calculates the shapley value for each feature value in this instance by considering all feature combinations and those shapley values contributes to the final explanation.

4.3.2 Comparison between decision tree and subgroup discovery

For the purpose of recognition of patterns in dataset, data mining techniques are considered. In this part, we would like to observe the similarity and differences between two data mining techniques, which are decision tree algorithm and subgroup discovery technique. In principle, decision tree algorithm is considered as an interpretable model, whereas it could also be used to mine local patterns through the decision path, where each path is traversed from the root node to a leaf node. Since we desire to observe the pattern in data where the inspected variable has significant influence, we should use the impact of that feature as the label for each instance. And this label is a numeric value which could be measured by the local interpretation methods. As for the subgroup discovery technique, it is aimed to discover interesting patterns from the data with respect to the target. In this case, the target is the impact value of the feature.

4.3.3 Case Study

As claimed at the beginning, this interpretation framework supports explaining model-agnostic black box models on tabular data and textual data. In this part, we would like to show two case studies and conduct experiments on tabular data and textual data respectively.

Case study 1: Adult Income dataset

As described, in Adult Income dataset, there are 14 descriptive features and more than 40 thousand instances. And the label indicates whether a person earns more than 50K dollars per year or not. Of course, the first step is to process the dataset when it is already available. In this case, for the convenience of training black box model, label encoding technique was applied to those categorical features. To have a better understanding of those features, the feature importance was measured. Even though there are many approaches to measure the feature importance, in this thesis, two methods were mainly used. One way was to calculate the permutation feature importance score and rank by the score. Another idea was to computer the sum of the absolute shapley values for each feature, and the ranking order could be observed from the corresponding summary plot.

Next, we would like to inspect the influence of variable "Sex". We use three approaches to measure the impact of attribute "Sex". Firstly, we calculate the prediction change of attribute "sex" by binary flip approach, in this way, we could get a value indicating the influence for each instance. Secondly, by fitting a linear model through LIME, we could identify the weight for feature "Sex", which implied degree of impact. Lastly, the contribution of attribute "Sex" was estimated by shapley value by computing the marginal effect while considering all combinations of feature values.

Subsequently, after obtaining the impact score for feature "Sex" by various methods, subgroup discovery technique was applied to the dataset under the condition that

taking the contribution value of feature "Sex" as the target. Afterwards, several interesting patterns could be discovered.

Case study 2: Amazon review dataset

5 Results and Discussion

Results...

5.1 Experiment Results

5.1.1 Local interpretation methods comparison

As was pointed out before, in this experiment, the Breast Cancer dataset was examined. After exploring the dataset, it was noticed that the attribute "area_mean" was one of the most important features according to the permutation feature importance ranking. Without loss of generality, we could randomly choose one instance and try to provide reasonable explanations. In this scenario, we decided to choose instance 10 from the dataset, and the boosting algorithm was utilized to train a black box model. Therefore, the following results were demonstrated based on the experiment that was intended to inspect the variable "area_mean" in an ensemble model on the processed dataset.

First local interpretation approach was numeric perturbation, which was applicable only to numeric attribute. The assumption was that we already had the black box model and the selected instance being explained. It could be easily estimated that there was 89

The next local interpretation method was LIME. After selecting the instance of interest, it was required to generate neighborhood points by perturbing the instance. To be more specific, the neighborhood points were produced by sampling from Normal(0, 1), multiplying by the standard deviation and adding back the mean value. Then, a weighted intrinsic interpretable model was fitted to explain the prediction making by the classifier. In this way, the model was supposed to be locally faithful around the explained instance. As displayed in Fig 3, the classifier determined that the cancer was malignant with probability 0.89. It could be noticed that features, such as "area_mean" or "texture_mean", had huge impact on the prediction due to the large coefficient in the model. Note that this approach would discretize numerical feature values into quartiles. For instance, it was observed that when the condition "area_mean > 0.27" was met, the weight for this feature was 0.35. It could be interpreted that if we decreased one unit on this feature value, the chance of the cancer to be malignant would decline 0.35. And on the right side, the feature value pairs of the instance to be explained was displayed in a table format. The feature columns showed the feature names and the value column displayed the original value for the corresponding feature.

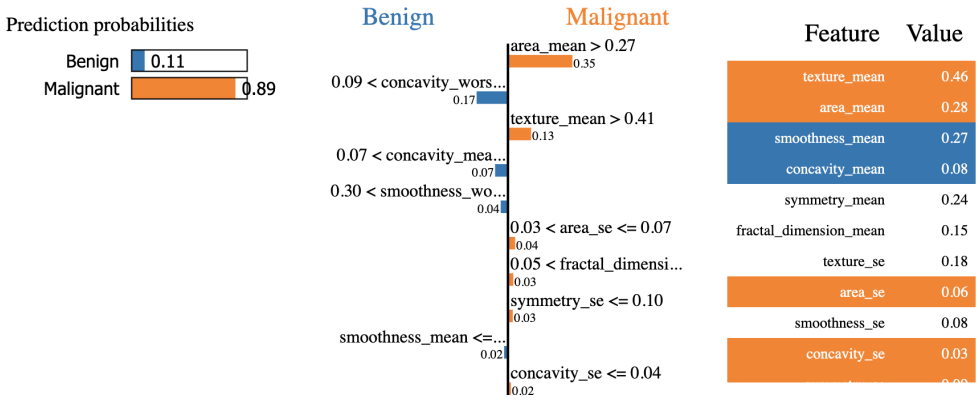

Figure 3: On the left side, the prediction probabilities were shown, telling that the tumor had 89% chance to be predicted as malignant. In the middle, the feature weights in the fitted model were displayed. In particular, the weight for attribute "area_mean" is 0.35. Note that the feature value pair of the instance being explained were listed on the right side.

Another local explanation method that we adopted was Kernel SHAP. Nonetheless, despite the fact that the back box model here we trained was tree ensemble model, the more efficient TreeSHAP estimation could be used instead. By applying this approach, it was aimed to compute shapley values for each feature, which were regarded as the feature contributions. As depicted in Fig 4 , it gave a nice reasoning which showed feature influence on this prediction. The shapley value could be visualized as "forces" and each feature value was a force either increased or decreased the prediction. As was seen, there was a base value denoted as -1.44, which was the average model output over all predictions. The below explanation showed features each contributing to push the model output from the base value to the actual model output. Features pushing the prediction higher were shown in red, those pushing the prediction lower were in blue. The shapley value for each feature was attached in the figure, but it had to be noted that by default the shapley value were displayed in the logit space and all those shapley values summed up to the difference between the model output for that instance and the expected base value. Therefore, we could infer that for this instance, attributes "area_mean" and "area_se" had a dominant effect on the prediction, while "concavity_worst" had reverse impact.

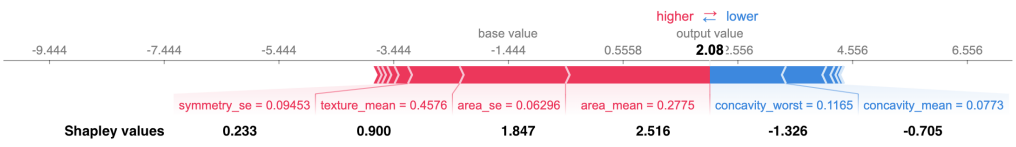

Figure 4: SHAP values to explain the influence of features leading to the prediction of benign or malignant cancer. The base value was the average probability over all predictions in logit space, which was -1.44. Shapley values were attached for each feature value as well. Feature values shown in red had positive effect in increasing the risk of being malignant cancer, while feature values denoted in blue declined the probability.

5.1.2 Decision tree vs. Subgroup discovery

Given the fact that both decision tree algorithm and subgroup discovery technique could be used to discover hidden pattern in data with respect to the effect of a specific attribute, it could be interesting to have a comparison. In this experiment, we decided to choose German Credit dataset, with the aim to predict whether a person has a good or bad credit risk. Since this dataset contained multivariate attributes, the dataset was processed by dummy encoding in this scenario. After exploring the dataset, the attribute "Credit amount" was selected to inspect the effect in the dataset. Thus, the first step was to measure the impact of feature "Credit amount" in each individual instance. In principle, all those aforementioned local interpretation methods supported the feature effect measurement. Nevertheless, Kernel SHAP approach was finally chosen. Following this approach, the shapley value of attribute "Credit amount" was computed for each instance.

By taking the shapley values as the target, it was assumed that local patterns could describe the influence of the selected attribute in the dataset. In this case, a tree-like graph could be drawn based on the decision tree algorithm. It had to be noticed that the maximal tree depth was set to three. As demonstrated in Fig 5, the first splitting node was "Purpose=furniture/equipment". By following the rightmost path, it led us to a pattern with high value, meaning that the attribute "Credit amount" revealed a significant impact on the dataset complied with this pattern. This pattern could be described as "Purpose=furniture/equipment AND Duration >13.5 AND Age > 31.5".

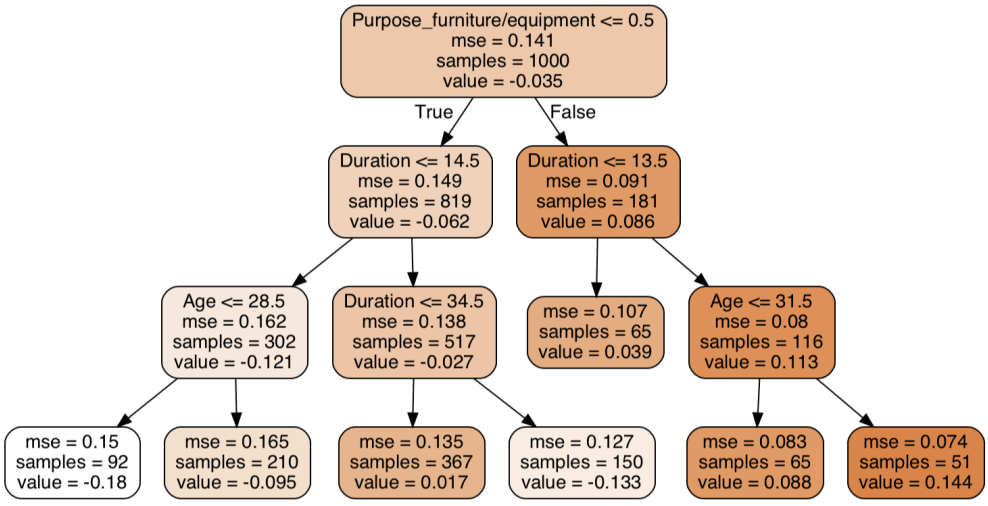

Figure 5: Decision tree

In contrast, by considering the shapley value as the numeric target in subgroup discovery technique, those interesting subgroups could be found out as shown in Table 6. As could be seen, the average shapley value over the entire dataset was about -0.03, while this value was 0.08 in the subgroup "Purpose=furniture/equipment", which also corresponded to the first splitting node. And the patterns "Duration >13.5"

and "Age $>$ 31.5" depicted in the path along the decision tree was also covered by the discovered subgroups in the table.

	quality	subgroup	size_sg	mean_sg	mean_dataset	mean_lift
0	0.015590	Purpose=furniture/equipment	181.0	0.086134	-0.034825	-2.473365
1	0.009869	Duration: [18:24[	153.0	0.064505	-0.034825	-1.852284
2	0.007420	Duration: [24:30[AND Job=2	133.0	0.078602	-0.034825	-2.257100
3	0.006187	Age: [33:36[	105.0	0.058926	-0.034825	-1.692074
4	0.004586	Duration: [24:30[	201.0	0.022816	-0.034825	-0.655160

Figure 6: Subgroup discovery results

Considering the similar patterns discovered by both techniques, we might argue that there indeed existed some patterns that the feature had a significant impact, and further influenced the model prediction.

5.1.3 Case study

As clarified in the last chapter, the first case study was experimented on the Adult Income dataset. The task for the experiment was to interpret the influence of a specific attribute in a black box model, and we assumed that the dataset and the model was provided. Initially, the dataset was processed as before by encoding the data and removing correlated features. Afterwards we randomly split the dataset into training set and testing set, and a simple full-connected neural network was trained based on the training data. By far, we had the neural network model and testing data, and the next step was to find out impact of a particular feature in a single instance. Furthermore, we were asked to discover some subgroups where the inspected feature had a dominant influence.

At the beginning, the importance of each feature was explored, which gave an indication about features that were worthy of attention. Apart from the permutation feature importance approach to estimate the importance degree of each feature, an alternative method to measure the importance was based on the magnitude of feature contributions using shapley values, called SHAP feature importance measure. The intuition was that features with large mean absolute shapley values were important. From the Fig 7, we could tell that "age" was the most important feature comparative to others. Considering that we would like to further compare the pattern mining results based on the feature effects calculated by binary flip approach and kernel SHAP approach, the attribute "sex" was chosen to explore since the binary flip method was designed for exploiting the binary feature.

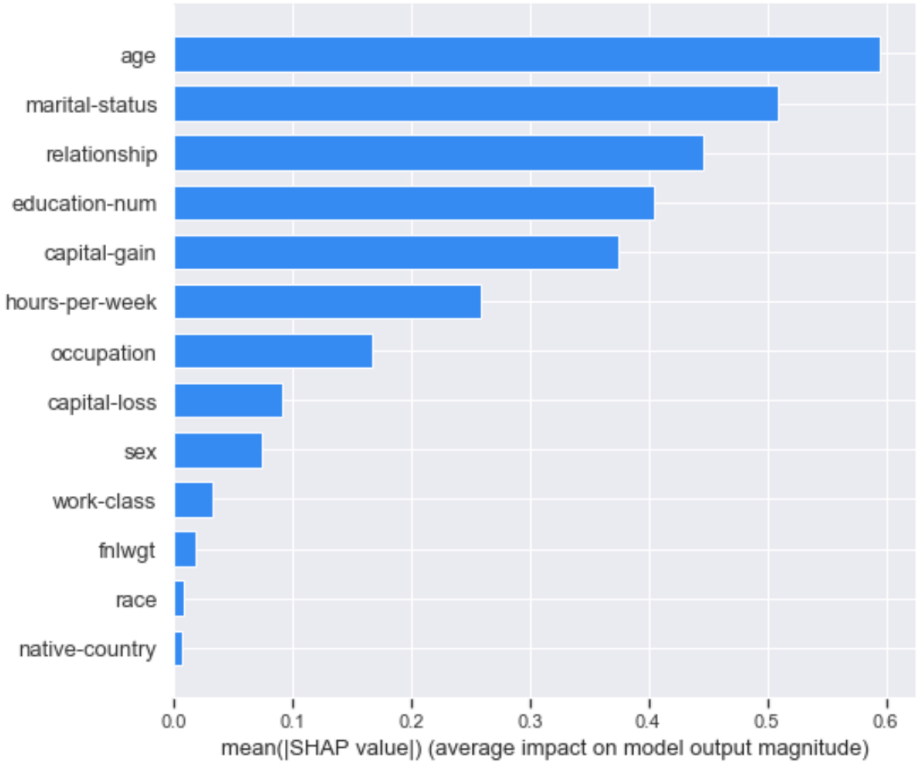

Figure 7: SHAP feature importance measure was based on the magnitude of feature contributions, and it was assessed by the mean absolute shapley values. "Age" was the most important feature.

From binary flip method, we could obtain the effect when changing the sex for each instance by calculating the prediction change. As for the kernel SHAP approach, the effect of sex was estimated by this feature contribution by computing the shapley values. Then subgroup discovery technique was applied to the dataset while taking the effect of sex as the target. Therefore, two different results were combined with respect to two different measurements and they were displayed in Table 8. Each part represented the interesting subgroups that were discovered. From Table 8, it was noticed that these two results were similar in some degree. For instance, they both discovered the pattern such as "relationship=Husband" and "marital-status=Married-civ-spouse".

	quality	subgroup	size_sg	mean_sg	mean_dataset	mean_lift
0	0.008143	relationship= Husband	3907.0	0.046788	0.026427	1.770480
1	0.006930	marital-status= Married-civ-spouse	4452.0	0.041632	0.026427	1.575385
2	0.003183	education-num>=13	2378.0	0.039502	0.026427	1.494769
3	0.002358	age: [50:58[	1165.0	0.046196	0.026427	1.748101
4	0.002046	age: [45:50[	997.0	0.046478	0.026427	1.758744

	quality	subgroup	size_sg	mean_sg	mean_dataset	mean_lift
0	0.022667	relationship= Husband	3907.0	0.056676	-0.005183	-10.934602
1	0.019314	marital-status= Married-civ-spouse	4452.0	0.042381	-0.005183	-8.176634
2	0.005462	occupation= Craft-repair	1211.0	0.044063	-0.005183	-8.501262
3	0.002567	hours-per-week>=55	1084.0	0.023130	-0.005183	-4.462564
4	0.002315	occupation= Transport-moving	514.0	0.043999	-0.005183	-8.488789

Figure 8: Subgroup discovery results

It could be concluded that there were patterns in data that feature “sex” had a huge impact no matter what local interpretation methods were used.

6 Conclusion and Future work

conclusion and future work...

6.1 Conclusion and Feature work

6.1.1 Factors to consider

Conclusion: Local surrogate models, with LIME as a concrete implementation, are very promising. But the method is still in development phase and many problems need to be solved before it can be safely applied.

6.1.2 Summary

6.1.3 Outlook

References

[1] B. Kim, R. Khanna, and O. O. Koyejo, "Examples are not enough, learn to criticize! criticism for interpretability," in *Advances in Neural Information Processing Systems*, 2016, pp. 2280–2288.

[2] W. J. Murdoch, C. Singh, K. Kumbier, R. Abbasi-Asl, and B. Yu, "Interpretable machine learning: definitions, methods, and applications," *arXiv preprint arXiv:1901.04592*, 2019.

[3] P. Voigt and A. Von dem Bussche, "The eu general data protection regulation (gdpr)," *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, 2017.

[4] D. Alvarez-Melis and T. S. Jaakkola, "On the robustness of interpretability methods," *arXiv preprint arXiv:1806.08049*, 2018.

[5] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 618–626.

[6] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 3319–3328.

[7] M. T. Ribeiro, S. Singh, and C. Guestrin, "Model-agnostic interpretability of machine learning," *arXiv preprint arXiv:1606.05386*, 2016.

[8] C. Molnar, *Interpretable Machine Learning*, 2019, https://christophm.github.io/interpretable-ml-book/.

[9] A. A. Freitas, "Comprehensible classification models: a position paper," *ACM SIGKDD explorations newsletter*, vol. 15, no. 1, pp. 1–10, 2014.

[10] M. Du, N. Liu, and X. Hu, "Techniques for interpretable machine learning," *arXiv preprint arXiv:1808.00033*, 2018.

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

[12] C. Strobl, A.-L. Boulesteix, A. Zeileis, and T. Hothorn, "Bias in random forest variable importance measures: Illustrations, sources and a solution," *BMC bioinformatics*, vol. 8, no. 1, p. 25, 2007.

[13] A. Fisher, C. Rudin, and F. Dominici, "Model class reliance: Variable importance measures for any machine learning model class, from the" rashomon" perspective," *arXiv preprint arXiv:1801.01489*, 2018.

[14] M. T. Ribeiro, S. Singh, and C. 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*. ACM, 2016, pp. 1135–1144.

[15] M. Robnik-Šikonja and I. Kononenko, "Explaining classifications for individual instances," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 5, pp. 589–600, 2008.

[16] I. Kononenko *et al.*, "An efficient explanation of individual classifications using game theory," *Journal of Machine Learning Research*, vol. 11, no. Jan, pp. 1–18, 2010.

[17] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems*, 2017, pp. 4765–4774.

[18] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery in databases," *AI magazine*, vol. 17, no. 3, pp. 37–37, 1996.

[19] H. Cheng, X. Yan, J. Han, and S. Y. Philip, "Direct discriminative pattern mining for effective classification," in *2008 IEEE 24th International Conference on Data Engineering*. IEEE, 2008, pp. 169–178.

[20] G. Dong and J. Li, "Efficient mining of emerging patterns: Discovering trends and differences," in *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*. Citeseer, 1999, pp. 43–52.

[21] F. Herrera, C. J. Carmona, P. González, and M. J. Del Jesus, "An overview on subgroup discovery: foundations and applications," *Knowledge and information systems*, vol. 29, no. 3, pp. 495–525, 2011.

[22] M. Atzmueller and F. Lemmerich, "Fast subgroup discovery for continuous target concepts," in *International Symposium on Methodologies for Intelligent Systems*. Springer, 2009, pp. 35–44.

[23] F. Lemmerich, "Novel techniques for efficient and effective subgroup discovery," 2014.

[24] D. Leman, A. Feelders, and A. Knobbe, "Exceptional model mining," in *Joint European conference on machine learning and knowledge discovery in databases*. Springer, 2008, pp. 1–16.

[25] W. Klösgen, "Explora: A multipattern and multistrategy discovery assistant," in *Advances in knowledge discovery and data mining*. American Association for Artificial Intelligence, 1996, pp. 249–271.

[26] B. F. Pieters, A. Knobbe, and S. Dzeroski, "Subgroup discovery in ranked data, with an application to gene set enrichment," in *Proceedings preference learning workshop (PL 2010) at ECML PKDD*, vol. 10, 2010, pp. 1–18.

[27] P. Clark and T. Niblett, "The cn2 induction algorithm," *Machine learning*, vol. 3, no. 4, pp. 261–283, 1989.

[28] M. J. Zaki, "Scalable algorithms for association mining," *IEEE transactions on knowledge and data engineering*, vol. 12, no. 3, pp. 372–390, 2000.

[29] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," in *ACM sigmod record*, vol. 29, no. 2. ACM, 2000, pp. 1–12.

[30] S. Wrobel, "An algorithm for multi-relational discovery of subgroups," in *European Symposium on Principles of Data Mining and Knowledge Discovery*. Springer, 1997, pp. 78–87.

[31] M. Ancona. (2019) Deepexplain: attribution methods for deep learning. [Online]. Available: https://github.com/marcoancona/DeepExplain

[32] M. Ancona, E. Ceolini, C. Öztireli, and M. Gross, "Towards better understanding of gradient-based attribution methods for deep neural networks," *arXiv preprint arXiv:1711.06104*, 2017.

[33] S. S. Marco Tulio Ribeiro and C. Guestrin. (2019) Lime: Explaining the predictions of any machine learning classifier. [Online]. Available: https://github.com/marcotcr/lime

[34] S. M. Lundberg and S.-I. Lee. (2019) shap: A unified approach to explain the output of any machine learning model. [Online]. Available: https://github.com/slundberg/shap

[35] F. Wang and C. Rudin, "Falling rule lists," in *Artificial Intelligence and Statistics*, 2015, pp. 1013–1022.

[36] L. Shapley, "A value for n-person games," *Contributions to the Theory of Games*, pp. 31–40, 1953.

[37] M. Atzmueller, F. Puppe, and H.-P. Buscher, "Towards knowledge-intensive subgroup discovery." in *LWA*. Citeseer, 2004, pp. 111–117.

[38] S. Garcia, J. Luengo, J. A. Sáez, V. Lopez, and F. Herrera, "A survey of discretization techniques: Taxonomy and empirical analysis in supervised learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 4, pp. 734–750, 2012.

[39] M. Atzmueller, "Subgroup discovery," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 5, no. 1, pp. 35–49, 2015.

[40] H. Grosskreutz, M. Boley, and M. Krause-Traudes, "Subgroup discovery for election analysis: a case study in descriptive data mining," in *International Conference on Discovery Science*. Springer, 2010, pp. 57–71.

[41] J. R. Quinlan, "Induction of decision trees," *Machine learning*, vol. 1, no. 1, pp. 81–106, 1986.

[42] A. Zimmermann and L. De Raedt, "Cluster-grouping: from subgroup discovery to clustering," *Machine Learning*, vol. 77, no. 1, pp. 125–159, 2009.

[43] A. Asuncion and D. Newman, "Uci machine learning repository," 2007.

[44] D. Dua and C. Graff, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[45] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[46] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial," *Frontiers in neurorobotics*, vol. 7, p. 21, 2013.

[47] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. ACM, 2016, pp. 785–794.

[48] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," in *Advances in Neural Information Processing Systems*, 2017, pp. 3146–3154.

[49] T. Chen and C. Guestrin, "Xgboost: extreme gradient boosting," https://github.com/dmlc/xgboost, 2019.

[50] T. F. T. W. W. C. W. M. Q. Y. T.-Y. L. Guolin Ke, Qi Meng, "Lightgbm: Light gradient boosting machine," https://github.com/microsoft/LightGBM, 2019.

[51] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, no. 5, p. 206, 2019.