Example and Feature importance-based Explanations for Black-box Machine Learning Models

A. Adhikari

Technische Universiteit Delft

Delft University of Technology

Challenge the future

Example and Feature importance-based Explanations for Black-box Machine Learning Models

by

A. Adhikari

in partial fulfillment of the requirements for the degree of

Master of Science
in Computer Science

at the Delft University of Technology,
to be defended publicly on Thursday October 25, 2018 at 2:00 PM.

Student number:	4627199	
Project duration:	January 1, 2018 – October 25, 2018	
Thesis committee:	Dr. D. Tax,	TU Delft, supervisor
	Prof. dr. ir. M.J.T. Reinders,	TU Delft
	Dr C.C.S. Liem,	TU Delft
	Dr R. Satta,	TNO, supervisor
	Dr M.S. Fath,	TNO, supervisor

An electronic version of this thesis is available at http://repository.tudelft.nl/.

Abstract

Machine Learning (ML) is a rapidly growing field. There has been a surge of complex black-box models with high performance. On the other hand, the application of these models especially in high-risk domains is more stagnant due to lack of transparency and trust in these black-box models. There is a disconnect between the black-box character of these models and the needs of the users. A sub-field of explainable machine learning has emerged to fix this disconnect but it is still in its early stages.

In this thesis we have developed a new method called LEAFAGE that is able to extract an explanation for a prediction made by any black-box ML model. The explanation consists of the visualization of similar examples from the training set and the importance of each feature. Moreover, these explanations are contrastive which aims to take the expectations of the user into account.

Furthermore, we evaluated the ability of LEAFAGE to reflect the true reasoning of the underlying ML model. LEAFAGE performs overall better than the current state-of-the-art method LIME, on ML models with non-linear decision boundary. At last, we performed a user-study to evaluate empirically how useful example and feature importance-based explanations are, in terms of perceived aid in decision making, acceptance and measured transparency. It showed that example-based explanations perform significantly better than providing no explanation and feature importance-based explanation, in terms of transparency, information sufficiency, competence and confidence. However in terms of acceptance no significant differences were found between the different explanation types.

Preface

Ten months ago I started the journey to the magical, scary and relatively fun world of research. My supervisor David Tax, suggested the topic of making ML models explainable to the users. I had been working with ML for about two years, but I had not really thought about that topic. I immediately liked the topic as I saw myself as the person who will save humanity from the dark lord ML models by making them understandable. I worked on this topic as a graduation intern in TNO. This journey taught me a lot about the challenges and possibilities for a harmonious interaction between the technical ML side and the social nature of explanations.

Without the guidance and support of a number of people, this thesis would not be possible. First of all, I would like to thank my supervisor David Tax. The most important thing that I have learned from him was to pursue research with a critical attitude and truthfulness. Through countless discussions I learned how to tackle different research problems. Second, I would like to thank my supervisors Riccardo Satta and Matthias Fath for their daily guidance and support. I found our brainstorm meetings very helpful, which always came with a lot of laughter. Furthermore I would like to thank my colleagues Martin, Marcel, Jasper, Maarten and Jurriaan of the Applied AI project of which my thesis was a part of, for interesting discussions and feedback. Also, many thanks to my colleagues of the Data Science department for their daily dose of humour and fun. I enjoyed our daily routine of playing table football during the breaks.

Furthermore, I would like to express my sincere gratitude and appreciation to my parents for their support through out my life and this thesis. They inspired my to pursue and excel in the things that I love to do. At last, many thanks to my friends and family for their encouragement and interest in my study. Special thanks to Jeanne for helping me through all the low and high points during the thesis.

A. Adhikari
Den Haag, October 2018

Contents

1 Introduction

As humans we are able to learn from our own experience and make informed decisions. Applying the same strategy to machines has resulted in the currently rapidly growing field of Machine Learning (ML): making data driven predictions. As machine learning models become more accurate, they typically become more complex and uninterpretable by humans. The black-box character of these models holds back its acceptance in practice, especially in high-risk domains where the consequences of failure could be catastrophic such as health-care or defense. Providing understandable and useful explanations behind ML models or predictions can increase the trust of the user. Furthermore, providing explanations allows to evaluate whether the black-box models make fair and ethical decisions i.e. that individuals are not treated unfairly because of their membership in a particular group.

The solution of making Artificial Intelligence (AI) understandable, such that its acceptance in practice becomes more feasible, is an emerging research topic namely eXplainable Artificial Intelligence (XAI) and it's biggest subdivision eXplainable Machine Learning (XML).

1.1. Case for Explainable Machine Learning

Humans will not take a predictive model seriously if they do not trust it. As ML models are increasingly being used in practice and influence our daily lives, it becomes crucial to evaluate their behaviour. To be trusted they should make decisions fair and ethically. The requirements for trust can differ subjectively. Trust in ML systems can be categorized in two granularity's as seen across literature [1, 2]:

- Trust in a model: indicates whether the ML model can be trusted to perform well after deployment. This trust can be obtained by evaluating a global explanation of the model that states the overall logic. Such explanations could answer questions such as "How does the model make predictions?", "In which situations does the model perform good/bad?" and "Which input features does the model deem important for its predictions?". For example an elected official, could have these questions when he/she has to approve the use of a ML model by the police for profiling criminals.

- Trust in a prediction: indicates whether a prediction can be trusted enough to base a decision on. A local explanation illustrating the reasons behind a prediction is needed to obtain this trust. Such explanations could answer questions such as "What is the logic behind this prediction?", "Why is this instance predicted as class A instead of class B?" and "Which input features were the most important behind this prediction?". For example, a doctor could have these questions when he/she is consulting a ML model for the diagnosis of a patient.

The need for explanations to obtain the two types of trust and fair and ethical decision-making is discussed in the following sub-sections.

1.1.1. Trust in a model

In many applications of ML, providing explanations of a prediction in run-time is not necessary, such as classifying an email as spam/non-spam or making internal decisions in a self-driving car. Nevertheless, it is crucial that the model works well in run-time. Typically, in the evaluation of ML models, cross validation is used to compute traditional performance metrics such as accuracy, area under the curve and confusion matrix which should approximate the performance of the classifier on unseen data. However, blindly relying on this approach has some drawbacks.

First, there could be a mismatch between optimized metrics and the actual metrics of interest. For example in user recommendation the target is to show interesting and engaging content to make the user stay and return. This target is an abstract concept and hard to quantify exactly. Using the amount of clicks on different articles to deduce partly how interesting it is, sounds reasonable and a ML model can be built to optimize that. If only traditional evaluation metrics are used, the performance might be high but "click-bait" articles will be given high importance which does not match the real target of user retention [1].

Second, sometimes there might be unintentional leakage of information about the target variable into the training and validation set that would not be present when deployed [3, 4]. An example is given in [4], in which a ML model has to predict whether a user of a retail website will leave their website or stay to view another page. Unintentionally there was a variable session length that stated the number of total visits by the user. This variable would give away the target and the obtained model would not be useful in practice. This leakage could be avoided by for example investigating the legitimacy of the discriminative features that predict the target variable [3].

Third, data shift occurs when the training and the test contexts do not match [5]. For example in [1] they trained a classifier to predict whether an emails content is related to Christianity or atheism. The trained model obtained an accuracy of 92% but their explanation framework of the model showed that the word "posting" was a significant feature which indicated atheism. It turned out that 20% of the atheism emails and only 2 emails of Christianity contained this word in the training and validation set, which would not be the case when deployed. Moreover, the deployment environment can be non-stationary. In some applications there could be adversarial agents trying to undermine the predictive power of the model.

Humans are good in generalization. We are able to transfer learned skills to unfamiliar situations [2]. By inspecting the logic of a model through explanations, an expert in the field, even without technical background, can assess whether the logic of the model can be generalized and avoid aforementioned pitfalls. Furthermore, the system designer can improve the model through feature engineering, parameter tuning or choosing another model [6, 7].

1.1.2. Trust in a prediction

In some applications, the learning system is used to provide useful information to human decision makers [1, 2]. In this case the users can combine their own knowledge and experience with the information given by the model to make the final decision. Providing useful explanation along with the prediction is crucial. The application of ML models in these use-cases is challenging, because of the lack of transparency of black-box models, An example of this use-case is helping a doctor to diagnose a patient. Doctors want to understand the reasons behind a prediction such that they can combine it with their own reasoning. The doctor will not trust the predictive model without explanations.

There already exists ML systems that decision makers use to ask for recommendations. IBM Watson is a famous example. One of its biggest product is Watson for Oncology, which provides treatment recommendations to doctors of patients with cancer. Given the medical details about the patient, Watson provides three sets of treatments namely recommended, for consideration and not recommended. According to the Watson for Oncology model, these set of treatments are respectively a good fit, a moderate fit and a bad fit for the patient. Along with each treatment it provides published articles and statistics backing up the treatment. The recommendation along with the evidence can help the doctor to make an informed decision.

STAT researched the use of Watson for Oncology in different hospitals around the world [8]. STAT reported that Watson isn't living up to their expectations and has a hard time to be adopted by hospitals.

One of the reasons for the distrust is the black-box character of the suggestions. Even though each of the suggested treatment is backed-up by evidence, Watson does not explain why the treatment is a good fit for the patient. Dr. Taewoo Kang, a surgical oncologist who used Watson said that "when asking Watson for advice on a patient whose cancer has not spread to the lymph nodes, and Watson will recommend a type of chemotherapy drug called a taxane. But that therapy is normally used only if the cancer has spread to the lymph nodes. And, to support the recommendation, Watson will show a study demonstrating the effectiveness of the taxane for patients whose cancer did spread to their lymph nodes." [8] In this case, the provided information is not useful to Dr. Kang as Watson cannot explain why the treatment is recommended even though it is normally not applied in the situation of his patient.

1.1.3. Fair and ethical decision making

Algorithmic decision-making influences more and more the daily lives of people [2]. It can range from personalized online advertisements to more serious encounters such as getting approval for a bank loan, filtering job applicants and accessing the likelihood whether a criminal defendant will re-offend in the future by the government [9]. It becomes crucial to make sure that there is no unfair treatment of individuals because of their membership in a particular group. This could be mitigated by verifying the logic behind a ML model and its predictions through explanations, but it is not always possible.

ML models need data to learn from and typically it is collected from society. The data will contain immoral bias to the extent that the society does [10]. The ML models will reproduce this biased logic for the classification if it is present in the training-data and it will be accurate when only traditional evaluation metrics are used.

Recently on 25 May 2018, a set of comprehensive regulations adopted by the European Parliament for the collection, storage and use of personal information, the GDPR (General Data Protection Regulation) took effect [11]. The GDPR prohibits automated processing based on the following special categories of sensitive data:

> personal data revealing racial or ethnic origin, political opinions, religious or philosophical beliefs, or trade-union membership, and the processing of genetic data, biometric data for the purpose of uniquely identifying a natural person, data concerning health or data concerning a natural person's sex life or sexual orientation ...

By removing sensitive variables from the input of the learning algorithm, one might think that the resulting model will not discriminate. This assumption has been widely disproved [12, 13]. It is possible that other variables correlate to the sensitive ones. For example, [13] showed that Facebook likes of a person could infer personal traits such as his/her race, gender, sex orientation, political view and religion with respectively an accuracy of 95%, 93%, 88%, 85% and 82%. These correlations might be very hard to detect by humans even through explanations especially in a large dataset with many features.

Furthermore, there could be uncertainty bias towards one group that is underrepresented in the training-set [11]. The predictions of an instance from a majority group can be made with a higher certainty as they are represented well, compared to a minority group. Algorithms that are risk-averse will prefer to only make predictions with high certainty, thus discriminating against the minority group.

Providing explanations behind algorithmic-decision making is also required due to regulatory constraints. The GDPR states in Article 13 and 14, that a person subject to algorithmic decision-making has the right to "meaningful information about the logic involved". It is not clear though what precisely is meant by that, what kind of explanations and in which depth is required [11].

1.2. Multidisciplinary research

XML is not only a technical science but also has a social part which is as much as important. It is important to understand what kind of explanations are useful to the user. The following questions are important to address to increase the trust of a user in a prediction. What do people expect from an

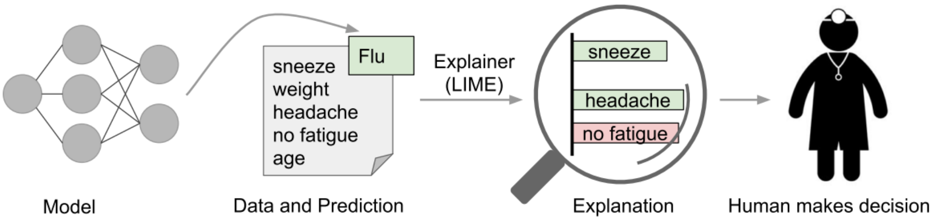


Figure 1.1: General overview of LIME [1]

explanation? How can an explanation efficiently help the user in decision-making? These questions have been extensively researched in psychology, philosophy and cognitive sciences. These sources are leveraged in this thesis.

One of the main findings is that, when people ask a why-question, they do not want to hear the whole causal explanation, rather they are interested in a subset (contrastive explanation) that can answer the conflict between the observed event and their own mental model of causation [14, 15]. In XML context a contrastive explanation answers why class A (the most probable class) was predicted instead of class B (the class that the user expected). Another important finding is that example-based reasoning is an effective strategy for tactical decision-making [16–18]. Example-based reasoning entails leveraging previous experience with analogous tasks to make a decision. At last, we consider the finding that people prefer a cause that is more responsible for the occurrence of an event over others. In XML context it can mean that more important features for a prediction are preferred over others.

1.3. Current solutions

There are three main strategies to extract human-understandable explanations from ML models, namely globally transparent, model-oriented and model-agnostic [15].

Globally transparent strategy tries to find a balance between being accurate while not too complex, such that the whole model is understandable to humans. In this case interpretable and transparent ML models such as decision trees and linear model are used (see section 2.2.2). For example, the feature weights of a linear model or the path that a prediction of a decision tree follows from the root to a leaf can be shown as an explanation. This strategy poses a severe limitation because in some cases complex black-box models have proven to perform better than globally transparent models [6].

In model-oriented strategy, the internal workings of the ML model is leveraged to extract an explanation. This strategy has the advantage that the explanations reflect the true reasoning of the ML model with high accuracy. The downside is that the explanation methods are model-dependent, requiring a different explanation method for each type of ML model.

At last in model-agnostic strategy, the ML model is seen as a black-box. The inputs and outputs of the model are analyzed to gain insights in the behaviour of the model. An advantage of a model-agnostic explanation method is that it can be used for any type of ML model. Further, the ML model can be built without taking explainability into account. A downside of these methods is that they can be less accurate than the other strategies in terms of reflecting the true reasoning of the ML model. LIME [1] is a well known model-agnostic method. An example of such method is provided in figure 1.1. The black-box classifier predicts that the patient has flu from the given symptoms. The explanation method LIME shows three symptoms that are the most important factors from the prediction as an explanation. "Sneeze" and "headache" contribute towards the prediction while "no fatigue" is evidence against it.

The focus of the current research in XML is heavily on the technical part of creating XML method. There is less focus on the human part of XML i.e. providing explanations according to the need of the users. Social research on explanations are generally not being leveraged in XML research [19].

1.4. Research questions

This study focuses on two types of explanations namely, feature importance-based and example-based explanations. Furthermore, we research on how to create contrastive versions of these explanations. The research questions entails:

1. *How can feature-importance and example-based explanations make the prediction of a complex ML model transparent?*
   We develop a new XML method called LEAFAGE (Local Example and Feature importance-based model AGnostic Explanation). It can provide a contrastive explanation for a prediction made by any black-box ML model. The explanation is local, i.e. it explains the reasoning of the black-box ML model in the neighbourhood of the instance being explained. More specifically, this explanation provides similar examples from the training-set and the importance of each feature for the prediction.
   (a) *What is the importance of each feature for a prediction made by a black-box ML model?*
   (b) *Which examples from the training-set are the most suitable as an explanation for the prediction made by a black-box ML model?*

2. *How can we make a feature-importance and example-based explanation, contrastive?*

3. *What is the quality of feature importance-based and example-based explanations extracted from LEAFAGE?*
   There are two main criteria for the evaluation of an explanation. Firstly, the local explanation should reflect the true reasoning of the underlying black-box ML model in the neighbourhood of the instance being explained. And secondly, the explanation should be useful to the user. We have developed a new evaluation method to measure the fidelity of a local explanation to the underlying model and have set-up an user-study.
   (a) *What should the scale of the evaluation be, which assess the local fidelity of an explanation?*
   (b) *How faithful is a LEAFAGE explanation to the underlying black-box model?*
   (c) *How useful are feature importance-based and example-based explanations to the user, in terms of the perceived aid in decision making, acceptance and measured transparency?*

1.5. Thesis Outline

This thesis is structured as follows. In chapter 2, background information and related work regarding XML is presented. It provides a human and technical view on explanation. First, we look at the preferred characteristics of an explanation from the perspective of the user by leveraging sources from social research. Second, the current XML methods that can extract explanations from ML models are shown.

Chapter 3 defines a new method called LEAFAGE. It also provides different examples that explain how the method works. This chapter aims to answer research question 1 and 2.

Chapter 4 aims to answer research questions 3.a and 3.b. It first defines an evaluation method to assess the fidelity of an explanation to the underlying model. Further, it applies that evaluation method to LEAFAGE explanations.

Chapter 5 describes the set-up and results of an empirical evaluation of LEAFAGE. This chapter aims to answer research question 3.c.

At last, chapter 6 concludes this thesis. It also presents some future directions for research.

2
Background and Related Work

2.1. Users' perspective on explanation

In the recent years there has been an increase in research in XAI and efforts have been made to create explanation systems to make the black-box ML models more transparent, but mostly from a technical perspective. In XAI it is also important to understand what the expectations and needs are from the perspective of the user, hence making it a multidisciplinary research including social science. Miller et al. [19] analyzed 23 papers of the related work section from the website IJCAI 2017 Explainable AI workshop, which can be considered as highly relevant papers compiled by the XAI community. They looked into whether the references included articles on explanation in social science and whether the evaluation of the explanation was based on data from human behavioral studies. They found out that only 4 papers referenced relevant social science papers while only one used it in its model. ML researchers and data scientists tend to build explainers from their technical perspective without asking questions such as: 'What do people expect from an explanation?' and 'How can an explanation efficiently help the user in decision-making?' These questions have been extensively researched in psychology, philosophy and cognitive sciences and should be leveraged in XAI. Furthermore, it is also important to conduct user studies on the intended audience to evaluate the real usefulness of the explanation system to the user.

In conclusion, by leveraging social science research one can understand the characteristics of a good explanation and by conducting user experiments one can verify whether the explanation system is useful in practice. In the following subsections the main findings from social research in the scope of this thesis are presented i.e. explanations are contrastive, example-based reasoning and explanations are selected. At last, different evaluation methods of explanations from the perspective of the user are discussed.

2.1.1. Explanations are contrastive

As humans we have a mental model of the world through our experiences and imagination. Our reasoning and imagination can fill in the blanks even when incomplete information is provided. When people ask a why-question often they do not want the whole explanation rather they are interested in a subset that can answer the conflict between the observed event and their own mental model of causation.

Most researchers in psychology, philosophy and cognitive scientist in this field agree that all why-questions are contrastive [15]. Miller [15] states "people do not ask why event P happened, but rather why event P happened instead of some event Q".

In this thesis we will use Lipton's [14] definition of the events P and Q being respectively the *fact* and *foil*. The fact is the event that occurred and the foil is the event that did not occur. For instance, a user in robot-human interaction asks a robot "Why did you open the door?" and it answers "because it is getting warm inside". This answer would not satisfy the user if he/she wanted to know why the robot did not open the window (the foil) instead; the explanation of the robot does not explain why the foil did not take place. The possible foils are all feasible events except opening the door such as "instead of turning on the air-conditioning", "instead of asking for permission first" or "instead of starting to dance". The provided explanation by the robot would be useful if the foil was "instead of leaving the door closed".

As humans we are good at extracting the implicit foil of questions from language, tone and context, moreover explanations are given, having a foil in mind [15]. It is challenging to detect the foil automatically by a machine. This is both a curse and a blessing because answering contrastive questions is often easier than providing a full explanation taking all possible causes into account . If the foil is clear, the machine only has to understand the difference between the two events to provide a satisfactory answer [14].

2.1.2. Example-based reasoning

In example-based reasoning the reasoner relates to previous experiences to understand and solve a current problem he/she faces [20, 21]. This type of reasoning lies very close to how we as humans think [22, 23]. We use it in our daily lives to solve problems. This learning approach has different names in the literature namely cased-based reasoning (CBR), example-based reasoning (EBR) and analogical reasoning (AR). In this thesis the term EBR is used.

The process of EBR can be typically divided into three steps namely retrieve, adapt and learn. For example, let's take a problem of choosing a dish to cook in this situation: no onions at home and wanting a light meal. We might think back to the different previous dishes we made and choose a few of them that suit the current situation (retrieve step). It is possible that the chosen dishes do not completely fulfill the requirements of the current situation e.g. all the known dishes use onions. In that case the most suitable known dish is picked and adapted (adapt step) e.g. use cheese instead of onions. The new dish can be leveraged to make future dishes or avoided completely (learn step), according respectively to the liking or disliking of the dish.

The applications of EBR can be divided into two types namely problem-solving and decision-justification [20]. In problem-solving previous similar situations are used as aid to decide how to proceed with the current situation. While in decision-justification previous similar situations are leveraged to support or dismiss certain possible decisions.

EBR for the purpose of problem-solving is commonly used in the health-care sector [16, 22]. For example, physicians think back to patients from the past that had similar symptoms as the patient they are examining. They remember the diagnosis of those previous patients and which treatment worked. That information helps them to diagnose the patient in front of them and to suggest a treatment. Moreover, real examples of medical cases in past are being used to train health-care professionals, complete guidelines and provide anecdotal accounts of treatments of individual patients in the medical literature [22]. Furthermore, EBR software systems are being used to aid health care professionals in retrieving similar medical cases from the past [16].

Law is a prominent domain in which EBR is used for the goal of justifying arguments, positions and decisions. Layers use EBR to justify a position by providing supporting relevant cases from the past [20]. Moreover, common law, which is widely used in most English-speaking countries, is based on precedence i.e. judicial decision made on similar cases from the past [21]. HYPO is an example of a system that can help lawyers in example-based reasoning. HYPO works as follows:

- First, the new case is analyzed and relevant factors are extracted.
- Second, similar cases from the past are retrieved based on the relevant factors.
- Third, each of the retrieved cases are divided into two sets according to the preferred position in the case i.e. whether they support or are against the position. The most relevant cases are chosen from both sets.

- Finally, cases from the support and the non-support set are used to respectively create an argument in favor of the preferred position and list possible counter-arguments that the opponent can pose. Cases from the support set are further used to counter the counter-arguments.

2.1.3. Explanations are selected

There can exist a complex causal chain of an event. When people explain why an event has happened, they typically select a small subset of causes from the whole causal chain. Miller et al. (2017) [19] compiled the following six main criteria that people typically use to select causes to provide as an explanation.

1. As argued in section 2.1.1 people ask contrastive why questions. In that cases causes that can explain why the fact event occurred instead of the foil event are preferred.

2. People prefer abnormal or unusual causes over typical or obvious causes. Hilton and Slugoski [24] give an example of two causes for the the explosion of the Challenger shuttle in 1989 namely, because of the presence of oxygen in the atmosphere and because of faulty seals. The first cause is obvious because oxygen is always present in the atmosphere around any shuttle. But the second cause is abnormal which is not present in most shuttles.

3. Intentional actions as cause is preferred over non-intentional actions [25]. For example, if someone died by being pushed from a building, the intention to murder is deemed more important than falling from a building as cause.

4. Necessary causes are preferred to sufficient causes [14]. For example, if cause a and either causes b or c are necessary for an event e to occur, than cause a is preferred over b and c. This is because cause a is individually necessary while b and c are not, for e to occur.

5. A cause that is more responsible for the occurrence of an event is preferred over others. To take an example from Chockler and Halpern [26]: 'if someone wins an election 11–0, then each person who votes for him is less responsible for the victory than if he had won 6–5'

6. Leddo et al. [27] conducted a study to establish the preference of people in goals, preconditions and conjunctions of preconditions and goals as causes. They found out that goals were preferred over preconditions and conjunctions over goals. For example for the event 'Fred went to the restaurant', explanation 'Fred was hungry and had money in his pocket' was preferred over 'Fred was hungry' and 'Fred had money in his pocket' was preferred the least.

2.1.4. Evaluation of explanations

The goal of an explanation system of a ML model is to be useful in practice for the intended users, hence conducting user-studies on the explanation system is important. In recommendation systems, extensive research has been conducted in designing user-studies which evaluate explanations that clarify why a certain item is recommended, from the user's point of view [28–32]. In recommendation systems, explanations increase the user trust and satisfaction, allow a quicker and easier search for relevant items and persuade the user to purchase the recommended item [28].

Before conducting a user-study it is important to understand what goals the explanation system tries to achieve. [28] defines seven goals for an explanation system, namely transparency, scrutability, trust, effectiveness, efficiency, persuasiveness and satisfaction which are respectively listed below:

1. Transparency entails whether the user is able to understand the logic behind why the ML model made a certain prediction.

2. By making a prediction transparent, the user could spot errors in the reasoning, thus a logical next step is to allow the user to correct the reasoning of the ML model.

3. Trust expresses whether the user has confidence that the ML model will work in practice.

4. Effectiveness expresses whether the explanation behind a prediction helps the user in better decision making.

5. Efficiency indicates the effort and time needed in making a decision given an explanation.
6. Persuasiveness entails whether the user is persuaded to follow the prediction of the ML model, given the explanation.
7. Satisfaction expresses the feeling of joy while using the explanation system.

All of the goals can be evaluated subjectively by asking for the opinion of the user [28]. Moreover, trust, persuasiveness and satisfaction are subjective by nature, and can only by evaluated subjectively. But, transparency, effectiveness and persuasiveness can also be evaluated/measured objectively, by testing the users whether they have understood the reasoning behind the recommendations [1], evaluating whether the users get better suited recommendations with explanations compared to without any explanations [31] and measuring the interaction time [30] or the number of interactions needed to find a satisfactory item, respectively.

2.2. Explanations in machine learning

This section looks at extracting explanations from ML models from a technical point of view. In subsection 2.2.1, a global view is presented of the different types of XML methods. Further, in sections 2.2.2 and 2.2.3 a detailed analysis of globally transparent models and a XML method called LIME is presented, respectively. At last, section 2.2.4 describes how other papers evaluate the fidelity of a local explanation to the underlying model. The last three sections are more related to LEAFAGE. LEAFAGE provides local explanations using a type of globally transparent model namely linear models and the basic concepts of LIME has been used in the creation of LEAFAGE.

Let $f : \mathcal{X} \rightarrow \mathcal{Y}$ be a black-box ML model that solves a classification problem, with $\mathcal{X} = \mathbb{R}^d$. $\mathcal{X}$ and $\mathcal{Y}$ are called the input space and the target or output space, respectively. Next, let $X = [\mathrm{x}_1, .., \mathrm{x}_n]$ with the corresponding true labels $y_{true} = [y_1, .., y_n]$ be the training-set that f was trained on. Further, let $X' = [\mathrm{x}'_1, .., \mathrm{x}'_n]$ with the corresponding true labels $y'_{true} = [y'_1, .., y'_n]$ be the testing-set that can be used to test the performance of f. $\mathrm{x} \in \mathcal{X}$ is called an instance, a sample or a data point and $y = f(\mathrm{x}), y \in \mathcal{Y}$ is called the predicted class or label of x. Further, $(a_1, ..., a_d)^T$ with $\mathrm{x} = (a_1, ..., a_d)^T$ and $\mathrm{x} \in \mathcal{X}$ are called the feature or attribute values of x. In this thesis the term ML model refers to a Machine Learning model that solves a classification problem.

2.2.1. Characteristics of XML

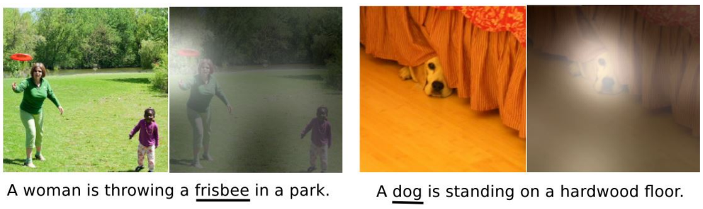
Global vs Local explanation

An explanation about a ML model can be of global or local scale. A global explanation clarifies the workings of a whole ML model. It explains how the relationship is modelled between the whole input space and the output space [33]. But one can also zoom into smaller regions of the input space and explain the working of the ML model there. These local explanations clarify the reasons behind the predictions of instances that fall into those smaller regions. In some cases the ML model can be very complex, thus making it hard to generate an understandable global explanation. But it is more likely that the logic of the ML model is much simpler in a small section of the input space, thus allowing to generate understandable local explanations.

An example of a decision tree model is shown in figure 2.2b. The classification problem is to classify a house as value *low* or *high*, given its *age* and *area*. The whole decision tree can be seen as a global explanation, because it explains how the features *age* and *area* of a house is used to determine its value. But if one is only interested to understand why a particular house is predicted with a certain value a local explanation is needed. For example a house with *age*=10 and *area*=30 is predicted as value *high* because its *age* is greater than 5 and its *area* is greater than 20.

Strategy

In the literature there are three main strategies that extract human-understandable explanations from ML models, namely globally transparent, model-oriented and model-agnostic [15]. In the first strategy, the ML model is optimized to be accurate but also simple enough to be understandable by humans (see section 2.2.2). While in the second and third strategy, the ML model is not required to be globally



A woman is throwing a frisbee in a park.

A dog is standing on a hardwood floor.

Figure 2.1: An example from a paper by Xu et al. [34]. The explanation shows the most important part of the image for the prediction.

transparent, instead human-understandable explanations are extracted from the complex ML model. The difference between the last two strategies lies in the type of information used for the explanation extraction.

In model-oriented strategy, the internal workings of the ML model is leveraged to extract an explanation. For example, Xu et al. created an explanation method that can explain why the machine learning model classified certain contents in an image. The explanation consists of an attention map that shows which part of the image was important for the prediction. Figure 2.1 shows two examples, in which a mask placed over the original images shows which parts of the images were important for the prediction of a frisbee and a dog. The inner-workings of a LSTM ML model is directly leveraged to build the attention map. This strategy has the advantage that the explanations reflect the true reasoning of the ML model with high accuracy. The downside is that the explanation methods are model-dependent, requiring a different explanation method for each type of ML model.

In model-agnostic strategy, the ML model is seen as a black-box. No internal information about the ML model is used, rather the black-box model is queried using a set of instances from the input space. The outputs of these queries are used to gain insights in the behaviour of the model, given the inputs. In some cases the training instances are used to query the black-box model [35] and in other cases new generated instances [1]. An advantage of a model-agnostic explanation method is that it can be used for any type of ML model. Further, the ML model can be built without taking explanability into account. A downside of these methods is that they can be less accurate than the other strategies in terms of reflecting the true reasoning of the ML model. LIME [1] is a well known model-agnostic method. LEAFAGE uses the basic ideas of LIME, hence it is important to have a good overview of LIME. In section 2.2.3 LIME along with another related method LS [35] is explained in detail.

2.2.2. Globally transparent models

In high-risk applications such as health-care and defense, globally transparent and interpretable model are preferred because it is important to understand how the ML model exactly works to avoid catastrophic consequences. The most prominent ML models that are regarded as transparent and interpretable in the literature [36], are listed below.

A decision tree partitions the input space into cuboid regions, whose edges are aligned with the axes [37]. Each cuboid region corresponds to a particular class. A new instance is predicted as the class of the region that it falls into. The process of getting the prediction can be described as the traversal from the root to a leaf of a decision tree. A two dimensional example of partitioning the input space and the corresponding decision tree, are shown in figures 2.2a and 2.2b, respectively. The classification problem is to classify a house as value low or $high$, given its age and area. First, the whole input space is divided into two region i.e. $age \leq 5$ and $age > 5$. Further, the left and right regions are divided into 4 sub-region according to $area \leq 15$ and $area > 15$ and $area \leq 20$ and $area > 20$, respectively. The predicted class of a new instance can be found by traversing from the top of the decision tree to a leaf e.g. a house with feature values $age = 7$ and $area = 15$ has the predicted class low. Decision trees are regarded as interpretable because predictions can be traced intuitively from the root to a leaf in a visual manner [38].

Decision rules are more general than decision trees. A decision rule maps an observation to an appropriate action. *If-then rules* are commonly used in which the *if clause* is a combination of conditions on an instance and the *then clause* contains an action. Some decision rules also contain an *else* clause that states which action to take if the *if clause* does not hold. The combination of conditions can be formed by conjunctions, negations and disjunctions. A decision tree can be converted into *If-then-else* rules. Figure 2.2c shows the example 2.2b converted into *If-then-else* rules. Decision rules are considered interpretable because they are can be presented in natural language [36].

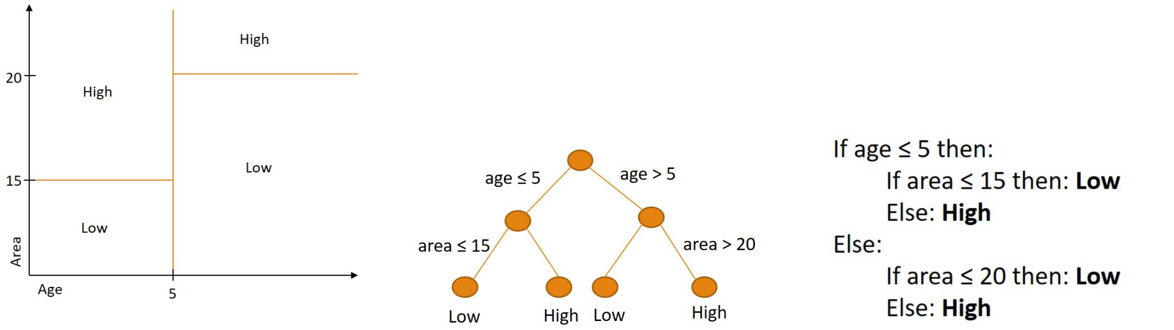
A Linear model can be used for a regression or binary classification problem. In a binary classification problem, a linear model aims to find a hyper-plane that separates the training instances according to their class values. A linear model can be written as:

$$y(\mathrm{x}) = w_0 + w_1 x_1 + w_2 x_2 + \ldots + w_d x_d$$

Generally the predicted label of an instance $\mathrm{x} = [x_1, .., x_n]$, is obtained by mapping all $y(\mathrm{x})$ values above a certain threshold to the first class c_1 and the rest to the second class c_2. The weights w denotes the most discriminative direction for the predictions made by the linear model. A positive (negative) sign of a weight w_i denotes a positive (negative) correlation between the increase (decrease) of x_i and the probability that x belongs to class c_1, given all other feature values stay the same. Moreover, the magnitude of w_i indicates the size of the increase or decrease of the probability. Furthermore, the magnitude of a contribution of a feature value x_i, to the prediction is equal to $abs(w_i * x_i)$.

When $d = 2$ of $d = 3$ the decision boundary can be visualized in a two or three dimensional plot, respectively. An example is visualized in figure 2.3, in which a linear model aims to separate *high* from *low* value houses, given its *area* and *age*. The first plot shows that the decision boundary is more aligned with the axis *area* than *age* i.e. feature *area* is more important for the classification then *age*. The second plot, illustrates the weights of the linear model. The absolute values of the weights is represented by the bar size and the sign by the colors. It shows that an increase in feature area and age, has a positive and negative contribution towards *high* value, respectively. Moreover, it also shows that the feature *area* ($w_{area} = 1$) is more important for the classification than feature *age* ($w_{age} = 0.5$). This can be seen as the global explanation of the linear model. Furthermore, the third plot provides a local explanation for the prediction of a house with *area*=40 and *age*=80. The bar size represents the importance of each feature value towards its prediction. The importance of *area* and *age* is $w_{area} * 40 = 20$ and $w_{age} * 80 = 40$ respectively. For the local prediction of this house, feature *age* is more important than *area* even though globally the opposite is the case.

These models have two drawbacks. Firstly, they are not inherently transparent and interpretable [2], despite their apparent simplicity. The size of these models can be enormous, making them complex and hard to grasp e.g. a decision tree model containing a lot of nodes or a long decision rules model or a linear model in a high dimensional space. They could be restricted to have a small size, but that can reduce the performance of these models significantly. Secondly, in certain domains complexer ML



If age ≤ 5 then:
If area ≤ 15 then: **Low**
Else: **High**
Else:
If area ≤ 20 then: **Low**
Else: **High**

(a) An two dimensional input space partitioned into 4 regions.

(b) A decision tree with two features.

(c) The decision tree converted into decision rules.

Figure 2.2: Example of a decision tree and its corresponding decision boundary and decision rules in a two dimensional space.

models such as deep neural-networks have proven to perform better than transparent models. In that case, choosing a transparent model will be at the cost of its performance.

2.2.3. LIME

LIME [1] stands for Local Interpretable Model-agnostic Explanations. Given the prediction of an instance $\mathbf{x}$ by a black-box classifier, LIME provides a local explanation which states why this prediction was made. It focuses on a small neighbourhood which is centered around $\mathbf{x}$. LIME approximates the decision boundary of the black-box classifier in this neighbourhood by an interpretable model and extracts an explanation from it. Linear regression model is chosen as interpretable model because the importance of each feature for the classification can be extracted from it.

The high-level pipe-line of LIME is shown in figure 1.1. The black-box classifier predicts that the patient has flu from the given symptoms. LIME provides an explanation in which the most important features are shown. The explanation shows that three symptoms are the most important factors from the prediction. "Sneeze" and "headache" contribute positively towards the prediction of having flu while "no fatigue" is evidence against it. The explanation can help a doctor to diagnose the patient.

First the general overview of LIME is described using the paper of LIME [1] together with the online available code [39]. Second, the shortcomings of LIME is analyzed.

Overview of LIME

We define $f_c : \mathbb{R}^d \rightarrow [0,1]$ as a black-box classifier and $\mathbf{x} \in \mathbb{R}^d$ as an input instance of f_c such that $f_c(x) = p$ with $p \in [0,1]$. d denotes the number of features and p the probability that $\mathbf{x}$ belongs to class c. LIME extracts an explanation for why the black-box classifier f predicted instance $\mathbf{x}$ as class c.

LIME needs three inputs namely the black-box classifier f, the training-set X which was used to build f and the instance that we want to explain $\mathbf{x}$. The workings of LIME can be divided into four steps:

1. First, artificial instances Z are sampled from a distribution modelled as uni-variate normal distributions of each feature in the training-set. The probabilities y given by f_c for the artificial instances Z are obtained. Furthermore, these Z instances are given weights according to their proximity to $\mathbf{x}$, by using an exponential kernel: $\pi_{\mathbf{x}}(\mathbf{z}) = e^{-D(\mathbf{x},\mathbf{z})^2/\sigma^2}$ with D a distance function and σ the kernel width. In short, LIME does not directly sample around the neighbourhood of x but samples new instances according to the distribution of the training-set and weights these artificial instances according to their proximity to $\mathbf{x}$. These weights represent the neighbourhood of $\mathbf{x}$ i.e. close by instances are given weights close to 1 and far away points close to 0.

   An illustration is presented in figure 2.4. The boundary of the black-box classifier is denoted by blue and pink background. The bold pink plus point is the instance being explained. The other pink and blue points are sampled artificially. The weights and the predicted labels of these artificial points are respectively represented by their size and color.

2. Second, feature selection is performed. The number of features d can be very high, showing the importance of all features can overwhelm the user. Thus only show a small k amount of features is used. Feature selection using the Lasso method is performed, i.e. k most discriminative features are chosen for the regression problem of Z with target values y. The weights of the artificial samples $\pi_{\mathbf{x}}(\mathbf{z})$ are used in the feature selection process, i.e. giving more importance to features that can separate the classes well in the neighbourhood of $\mathbf{x}$. Z containing only k local discriminative features is denoted as Z'.

3. Third, a linear model is trained to approximate the decision boundary in the vicinity of x. Let's define $g \in G$ where G is a class of all potential linear models. A locally weighted loss function L is used to get the optimal linear model $g*$. $g*$ is obtained by performing a least square regression.

$$L(f,g,\pi_{\mathbf{x}}) = \sum_{\mathbf{z}\in Z, \mathbf{z}'\in Z'} \pi(\mathbf{z})(f(\mathbf{z}) - g(\mathbf{z}'))^2$$

As illustration, figure 2.4 shows the linear model that approximates the local decision boundary around $\mathbf{x}$.

4. At last, an explanation is extracted from the linear model $g*$. An example of the explanation is given in figure 1.1.

LIME distinguishes between the original input space and the interpretable input space. The features in the interpretable input space are supposed to be understandable to humans. It suggests that a binary vector that indicates the presence or absence of the values of the instance being explained is interpretable. The interpretable input space differs per instance being explained. An example is shown in figure 2.5 in which for an instance being explained $\mathbf{x}$ with values *Age = 30* and *Area = 20* an interpretable input space is created that indicates whether an instance in the original input space falls in the same range as $\mathbf{x}$. These ranges are determined by discretizing the numerical features according to their distribution in the training-set e.g. discretizing *Age* in quartiles could be *Age < 5*, *5 ≤Age < 11*, *11 ≤Age < 20* and *Age > 20*. It further suggests to sample artificial instances (see step 1 algorithm above) in the interpretable input space and to further map it the original input space to get the predicted label. The linear model is than built in the interpretable input space. But it is not clear how it is possible to map from the interpretable input space to the original input space. When looking at the implementation, they sample in the original feature space and map that to the interpretable feature space.

Shortcomings of LIME

LIME chooses for a linear model as the interpretable model to approximate the local boundary. A big advantage of linear models is that the importance of the features can be extracted and provided as an intuitive explanation. However, it is possible that the linear model is too simple to represent a highly nonlinear local black-box decision boundary, i.e. a linear approximation will not grasp the logic behind the prediction well enough.

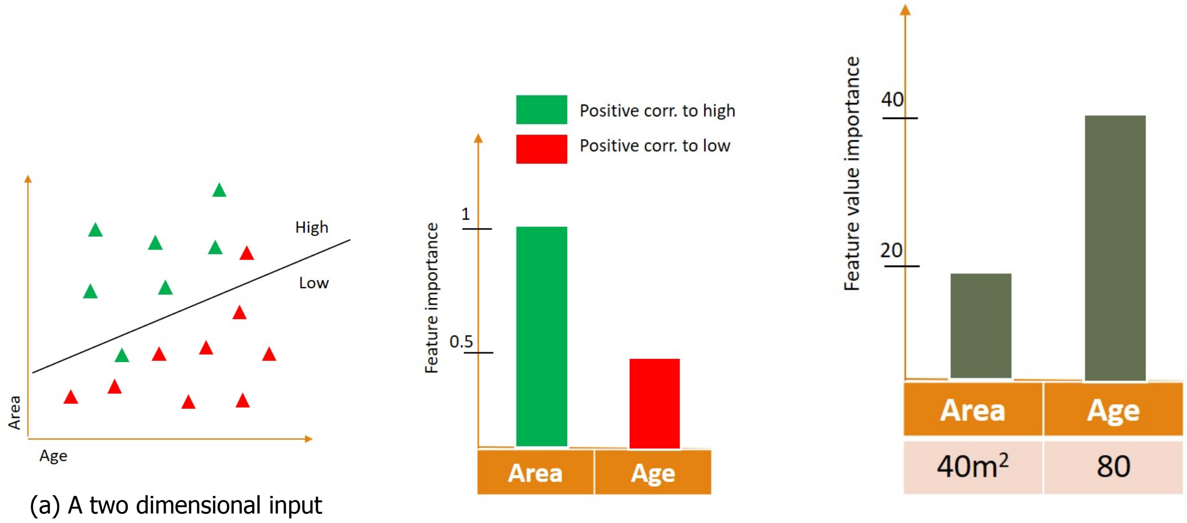
The approximation of the interpretable model is dependent on the instances used to learn from. We further analyze the sampling method used by LIME.

First, LIME chooses to sample artificial instances instead of using the training instances. It tries to sample realistic instances by sampling from a distribution modelled as uni-variate normal distribution of each feature in the training-set. This might not be realistic in some cases when the data does not follow a normal distribution.

Second, regardless of the first choice the size of the neighbourhood around x from which the instances are sampled is important. If an instance is very far from the decision boundary compared to a closer one, the neighbourhood should be bigger to incorporate the closest decision boundary. In LIME this neighbourhood is determined by σ and in its implementation it is set to $0.75 * \sqrt{d}$. σ is not dependent on the distance to the closest decision boundary but only on the dimension of the input space. This can lead to over-generalization or under-generalization of the local decision boundary respectively if the neighbourhood is set too big or small.

In figures 2.6a, 2.6b, 2.6c three important sampling scenarios are presented in which the size of the neighbourhood is varied by varying σ. It is a binary classification problem with red and green classes and with two input features. The decision boundary of the black-box classifier is defined as a parabola. The visualized points are instances from the training-set and the colors corresponds to the label given by the black-box classifier. The instance being explained is in yellow. In figure 2.6a the neighbourhood defined by $\pi(\sigma = 1)$ is too small in which all of the red instances have weights equal to zero. As a result the linear approximation will be arbitrary (under-generalization). In figure 2.6c the neighbourhood is too big which will result in a linear model that incorporates the global trends of the black-box model (over-generalization). The neighbourhood defined by $\pi(\sigma = 6)$ (figure 2.6b) is optimal in which enough weight has been given to instances from both class.

Laugel et al. [35] spotted the problem of over-generalization of the local decision boundary by LIME. They show an example (figure 2.7) in which the linear approximations (dotted lines) learned by LIME



(a) A two dimensional input space partitioned into 2 regions by a linear model. The triangles (color indicates class) are instances form the training-set.

(b) A global explanation of the linear model. The importance of each feature can be visualized in a bar plot.

(c) A local explanation of a prediction. The importance of each feature value can be visualized in a bar plot.

Figure 2.3: Example of a linear model that predicts the value of a house as high or low.

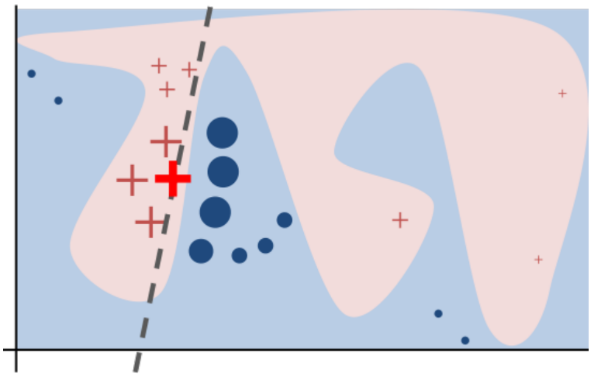
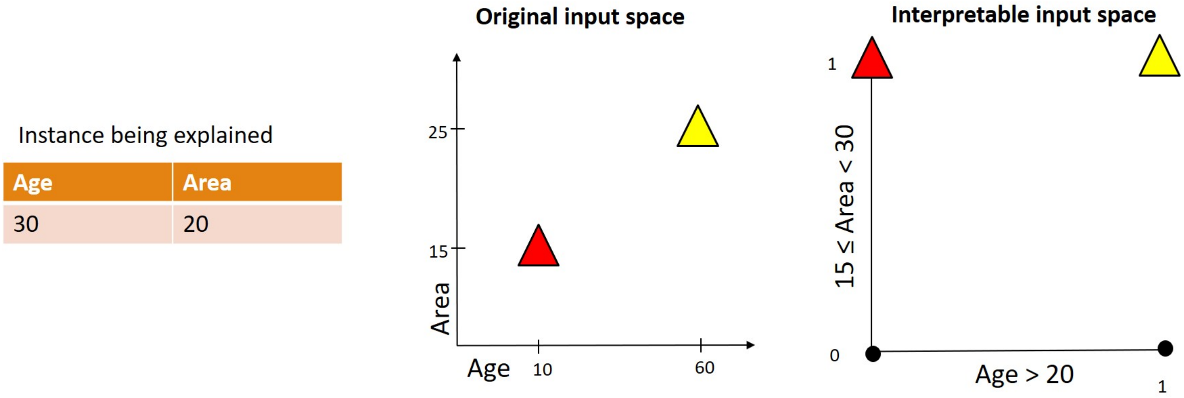


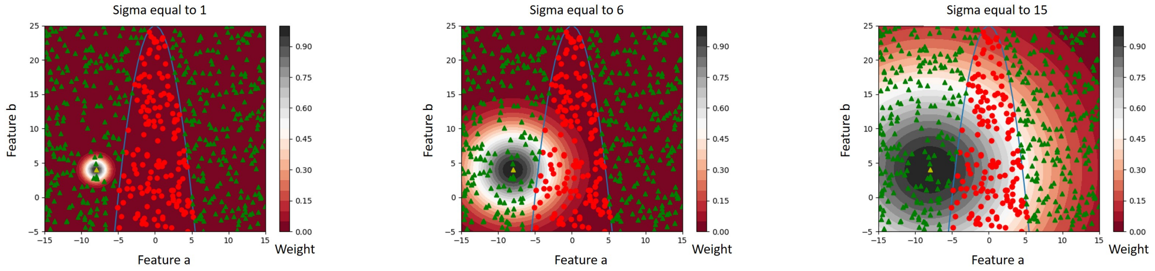
Figure 2.4: Overview LIME: Artificial instances are sampled around the bold plus point. A linear model is learned on them and used to extract an explanation. [1]



Instance being explained

Age	Area
30	20

Figure 2.5: An example of the mapping from the original input space to the interpretable input space.



(a) The neighbourhood defined by $\pi(\sigma = 1)$ is too small in which all of the red classes have weights zero.

(b) The neighbourhood defined by $\pi(\sigma = 6)$ is optimal in which enough weight has been given to instances from both class.

(c) The neighbourhood defined by $\pi(\sigma = 15)$ is too big in which the linear model tries to approximate the global model.

Figure 2.6: Three important sampling scenarios of LIME.

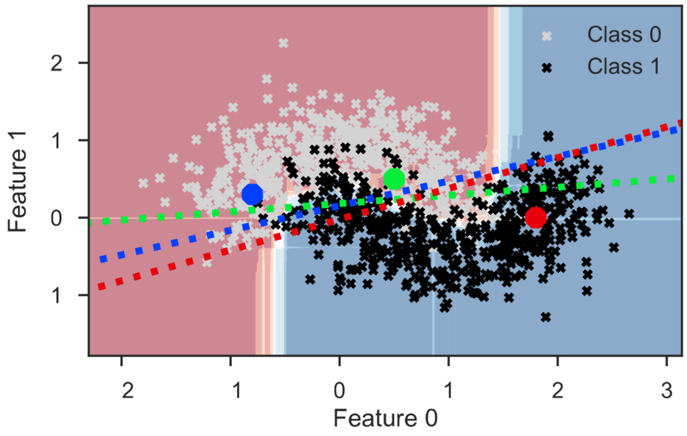
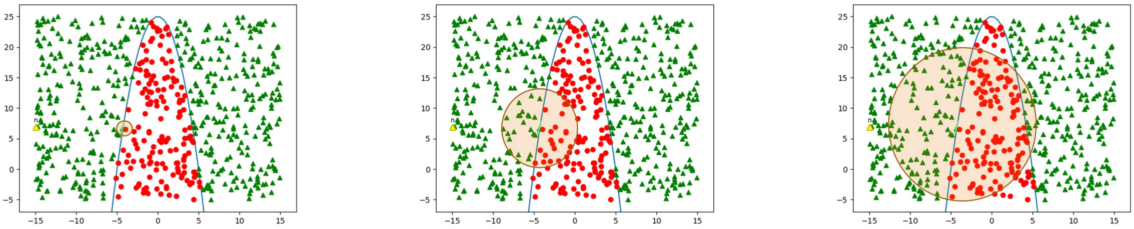


Figure 2.7: The linear approximations (dotted lines) are learned by LIME for the prediction of the 3 biggest colored points. The linear models of the points are very similar even-though the local decision boundaries are different. [35]

for the prediction of the 3 biggest colored points are shown. The black-box boundary is given by the blue and pink background. The linear models of the points are very similar even-though the local decision boundaries are different.

Laugel et al. [35] propose a new method LS (Local Surrogate) as an improvement on LIME and suggest to sample instances from the training-set within a hyper-sphere with the center equal to the closest decision boundary of x, and a radius r. r is fixed according to the maximum distance between the training instances. The closest decision boundary of x is approximated by using the growing spheres algorithm [40], in which artificial instances are sampled uniformly within a hyper-sphere of growing radius centered on n, until an instance from the opposite class n_{border} is found. The n_{border} is regarded as the approximation of the local decision boundary. This brute-force way of approximating the local decision boundary accurately is only feasible in a small dimensional input space but not in a high one, because the volume of hyper-sphere increases exponentially with the number of dimensions. Like LIME, LS also has three important sampling scenarios as shown in figures 2.8, in which the neighbourhood can be too small, have a good size or too big, respectively. When the neighbourhood is too small or too big, it will contain not enough training instances for a good linear approximation (figure 2.8a) or too many instances that are not linearly separable (figure 2.8c), respectively.



(a) The size of the neighbourhood is too small.

(b) The neighbourhood has a good size.

(c) The size of the neighbourhood is too big.

Figure 2.8: Three important sampling scenarios of LS

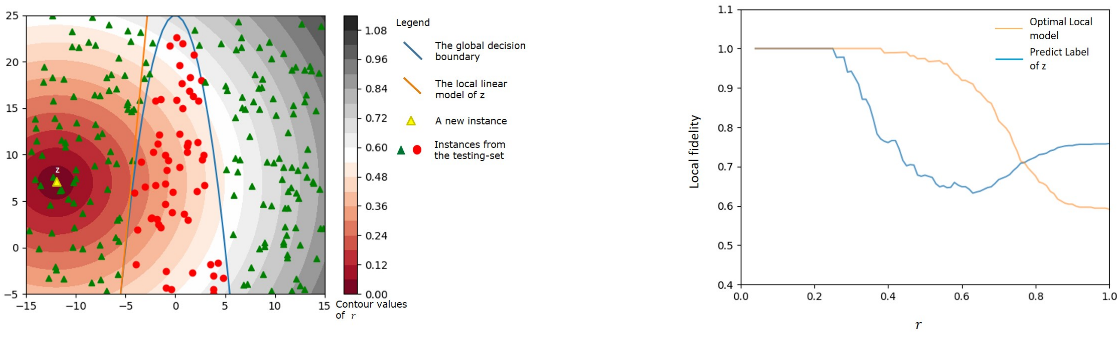
2.2.4. Evaluation of local explanations

An explanation that clarifies the local behaviour of a black-box ML model, should be faithful to the underlying black-box ML model. This subsection describes evaluation methods to measure fidelity of a local model-agnostic explanation (extracted from an interpretable model) to the underlying black-box ML model.

LIME [1] suggests to measure the fidelity as follows. Two interpretable models are chosen as black-box classifiers. They make sure that only 10 particular features (*gold* features) are used for any prediction by the black-box classifiers. Further, they generate an explanation that shows the 10 most important features for the prediction of each test instance. At last, they observe whether these important features contain the 10 gold features. This is a very soft method to measure the fidelity, because is does not test whether the relative importance of the features are respected in the explanation.

Method LS [35] defines a more rigid way to evaluate the local fidelity to the black-box ML model. It defines *the local fidelity* within a neighbourhood $\mathcal{V}_{\mathbf{z}\_r}$ of an instance $\mathbf{z}$ as an evaluation score (e.g. accuracy), of the predictions made by the interpretable model compared to f on the testing instances that lie in $\mathcal{V}_{\mathbf{z}\_r}$. $\mathcal{V}_{\mathbf{z}\_r}$ is defined as a hyper-sphere with $\mathbf{z}$ as center and $r \in [0, 1]$ as radius. r denotes the locality of the evaluation and is expressed as the percentage of the maximum distance between the instances of the testing-set and $\mathbf{z}$. The final evaluation-score of the testings-set is an average of the local fidelity scores applied on each of its instances. In the evaluation of LS, r is set to 0.05 for all instances, without any justification.

Figure 2.9a shows an example of a testing-set and the values of r applied on a two dimensional space. The corresponding local fidelity values (accuracy as evaluation score) of varying $\mathcal{V}_{\mathbf{z}\_r}$ over $r \in [0, 1]$ are shown in figure 2.9b. The blue and the orange line correspond to the local fidelity of the optimal linear model $g^*_{\mathbf{z}}$ and another model $b_{\mathbf{z}}$ that always predicts the label of $\mathbf{z}$, respectively. The local fidelity of $g^*_{\mathbf{z}}$ and $b_{\mathbf{z}}$ have optimal local fidelity score from 0 to 0.25 r. Furthermore, any arbitrary linear model that does not go through $\mathcal{V}_{\mathbf{z}\_r}$ with $r = 0.25$ and that predicts all the instances in the side of $\mathbf{z}$ as the label of $\mathbf{z}$, will have perfect accuracy. Thus in this case any local fidelity scores based on $\mathcal{V}_{\mathbf{z}\_r}$ with $r \in [0, 0.25]$ does not reflect how faithful the explanation is to the underlying model. The range of r in which the local fidelity scores are not valid depends on each instance. Instances that are far away from the decision boundary have a bigger range than instances that are closer.



(a) The contour values of r applied on a two dimensional example.

(b) Accuracy per radius of local linear model and a baseline model.

Figure 2.9: The local fidelity values with different values of r.

3
LEAFAGE

This thesis proposes a new method LEAFAGE that provides intuitive and understandable explanations for a prediction made by any black-box ML model. LEAFAGE stands for Local Example and Feature importance-based model AGnostic Explanation. The explanation makes the reasons behind a prediction transparent to the user, by providing examples from the training-set that are similar to the instance being explained and showing the importance of each feature for the prediction.

Let $f : \mathcal{X} \rightarrow \mathcal{Y}$ be a black-box ML model that solves a binary classification problem with $\mathcal{X} = \mathbb{R}^d$ and $\mathcal{Y} = \{c_1, c_2\}$. Let $\mathrm{z} \in \mathcal{X}$ be an instance of the input space with $f(\mathrm{z}) = c_z$, $c_z \in \mathcal{Y}$, for which an explanation is needed. Furthermore, let $X = [\mathrm{x}_1, ..., \mathrm{x}_n]$ with the corresponding true labels $y_{true} = [y_1, ..., y_n]$ be the training-set that f was trained on. Let $y_{predicted} = \{f(\mathrm{x})|\mathrm{x_i} \in X\}$ be the predicted labels of the training-set. Next, let $\{\mathrm{x} \in \mathcal{X} \ |f(\mathrm{x}) = c_z\}$ and $\{\mathrm{x} \in \mathcal{X} \ |f(\mathrm{x}) \neq c_z\}$ be defined as the ally and the enemy instances of z [40], respectively. LEAFAGE needs X, $y_{predicted}$, z and c_z to extract an explanation for why z was predicted as c_z instead of the opposite class.

In a high-level overview, LEAFAGE works as follows. First, a sub-set of the training-set in the neighbourhood of z, is used to build a local linear model, that approximates f in the neighbourhood of z. The importance of each feature for the classification of z is extracted from that linear model. Next, the importance of each feature is used to define a similarity measure that returns how similar an instance $\mathrm{x} \in \mathcal{X}$ is to z. This similarity measure is used to retrieve similar examples as z from the training-set. At last, the importance of each feature along with the most similar examples are given as an explanation for why f classified the instance z as c_z.

In subsections 3.1 and 3.2 the similarity measure and two strategies to build the local linear model are defined, respectively. Further, subsections 2.1.1 and 3.4 explain how an LEAFAGE explanation can be made contrastive and visualized, respectively.

3.1. Defining Similarity

A classification example is shown in figure 3.1, in which the black-box classifier predicts whether a house has a *high* or *low* value according to its *area* and *age*. A new house z is predicted as value *high*. To find similar houses a similarity measure has to be defined. A trivial solution is to use euclidean distance measure which gives equal weights to all features. Figure 3.1a shows two potential houses (x_1 and x_2) from the training-set. According to the euclidean distance, z is more similar to house x_1 than house x_2. But the black-box classifier only looks at the feature *area*. Thus, according to the black-box classifier z is more similar to x_2 than x_1 (figure 3.1b)

The importance of each features for the classification of z can be retrieved by approximating the decision boundary of the black-box classifier linearly as shown by the blue line in figure 3.1c. Let $g(\mathrm{x}) = \mathrm{w}_z\mathrm{x} + c$ with $\mathrm{w}_z = (w_{z1}, ..., w_{zd})^T$ be the linear model that approximates the decision boundary. w_z denotes the most discriminative direction for the classification of z.

In figure 3.1 the global decision boundary is a horizontal line which can be approximated accurately by a linear model. But in practice the decision boundary of the black-box ML model can be arbitrarily complex as shown in figure 3.2. In that case, a linear model will not be able to approximate the the global decision boundary accurately. However, we assume that a small fragment of the global decision boundary which is the closest to $\mathbf{z}$ is smooth enough to be linearly approximated accurately, as illustrated by the blue line in figure 3.2. In that case the linear model is not valid globally, but only locally in the neighbourhood of $\mathbf{z}$. We further assume that the closest decision boundary to $\mathbf{z}$ is the most important fragment of the global decision boundary for the classification of $\mathbf{z}$ [35, 40].

The following definitions describe the local behaviour of the black-box ML model around $\mathbf{z}$. These definitions applied to the housing example are illustrated in figure 3.2.

Definition 3.1. Let *the local decision boundary* of $\mathbf{z}$ be defined as the closest fragment (according to euclidean distance) of the global decision boundary to $\mathbf{z}$.

Definition 3.2. Let *the local linear model* of $\mathbf{z}$ be defined as the linear model that approximates the local decision boundary of $\mathbf{z}$.

Definition 3.3. Given the local linear model $g(x) = \mathbf{w}_z \cdot \mathbf{x} + c$ of $\mathbf{z}$ let the *the black-box similarity measure* between $\mathbf{z}$ and an instance $\mathbf{t} \in \mathcal{X}$ be defined as the following:

$$b(\mathbf{t}) = \sqrt{d} \cdot \left\|\mathbf{w}_z^T\mathbf{t} - \mathbf{w}_z^T\mathbf{z}\right\| + \|\mathbf{t} - \mathbf{z}\|,$$

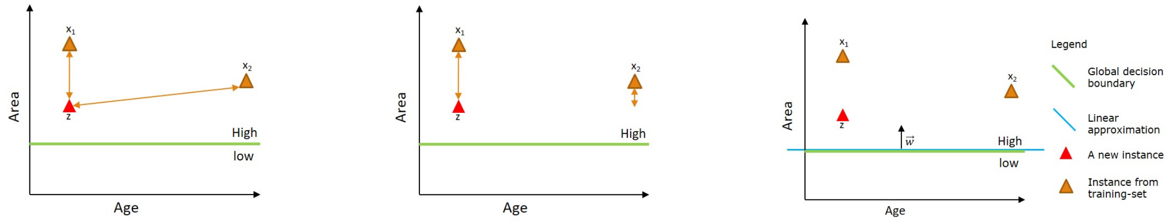
The black-box similarity values on the two dimensional housing example is shown on figure 3.3. In the first term of the black-box similarity formula, $\mathbf{z}$ and $\mathbf{t}$ are projected onto the direction $\mathbf{w}$ (extracted from g) and euclidean distance is applied onto those projected values, because $\mathbf{w}$ it is the most discriminative direction for the classification of $\mathbf{z}$. But g is only valid in the neighbourhood N of $\mathbf{z}$, and it is not straightforward to define N. This thesis proposes a heuristic solution in which the fact that closer instances to $\mathbf{z}$ (according to the euclidean distance on the input space) are more likely to be within N, is leveraged. The black-box similarity measure (definition 3.3) weights the euclidean distance on the input space and the euclidean distance on the vector $\mathbf{w}$ equally.

3.2. Computation of the local-linear model

Definition 3.4. Let *the local training-set* of $\mathbf{z}$ be defined as the instances $X_z = \mathbf{x}_1, .., \mathbf{x}_t$, $\forall \mathbf{x} \in \mathcal{X}$ with labels $y_l = \{f(\mathbf{x}) | \mathbf{x} \in X_z\}$, which are used to build the local linear model of $\mathbf{z}$.

For the generation an accurate local-linear model the sampling strategy of the local training-set of $\mathbf{z}$ is very important. Methods LIME [1] and LS [35] have proposed solutions for sampling instances, which are explained and reviewed in detail in section 2.2.3. We suggest three desired characteristics that a local training-set of $\mathbf{z}$ should adhere to, taking the shortcomings of LIME and LS into account:

1. The convex hull of the local training-set of $\mathbf{z}$ should contain the local decision boundary of $\mathbf{z}$.



(a) According to the euclidean distance $\mathbf{z}$ is more similar to $z1$ than $z2$

(b) According to the black-box classifier $\mathbf{z}$ is more similar to $z2$ than $z1$

(c) The black-box global decision boundary can be approximated accurately by a linear model.

Figure 3.1: Simple decision boundary of a black-box ML model that prediction whether a house has *low* or *high* value. A new house $\mathbf{z}$ is predicted as value *high*.

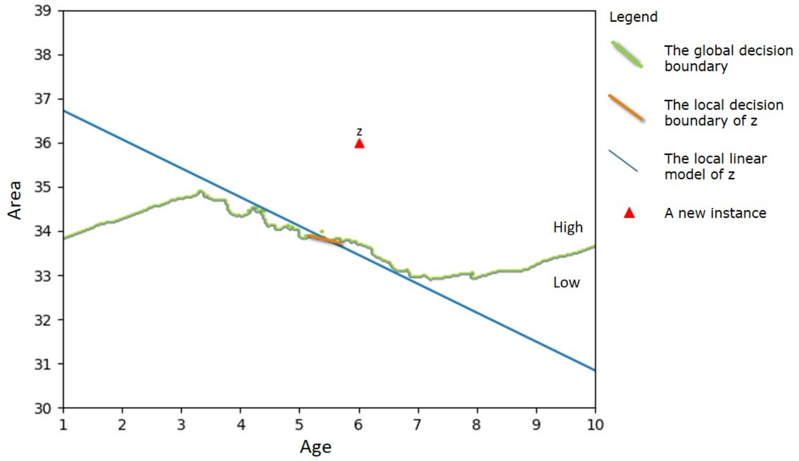


Figure 3.2: A complex decision boundary that cannot be accurately approximated by a linear model.

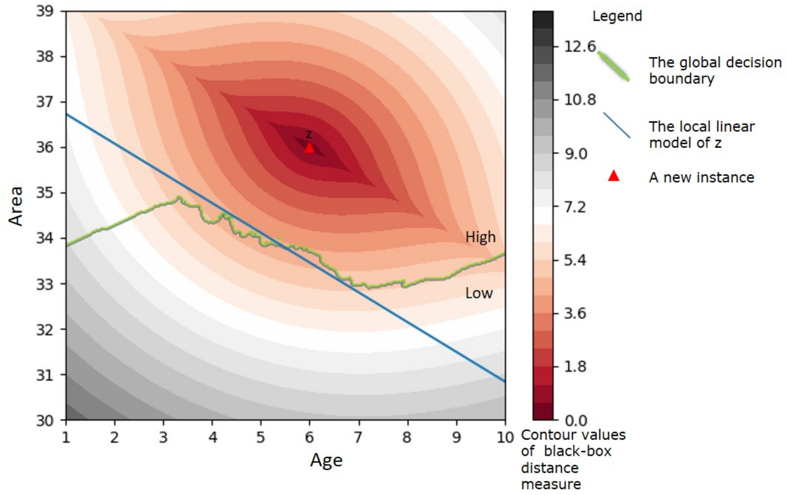


Figure 3.3: The contour lines of the black-box similarity measure (definition 3.3) applied on a 2D classification example

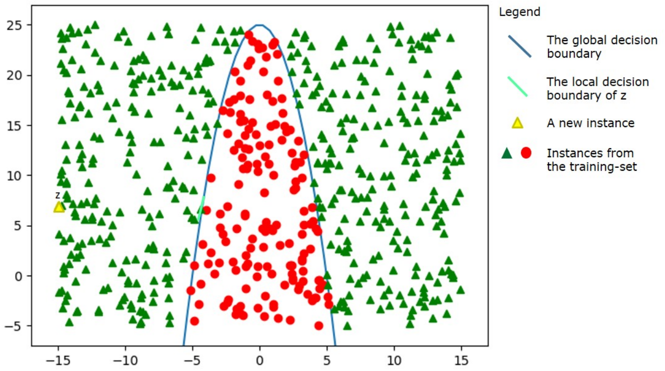


Figure 3.4: Two dimensional classification problem

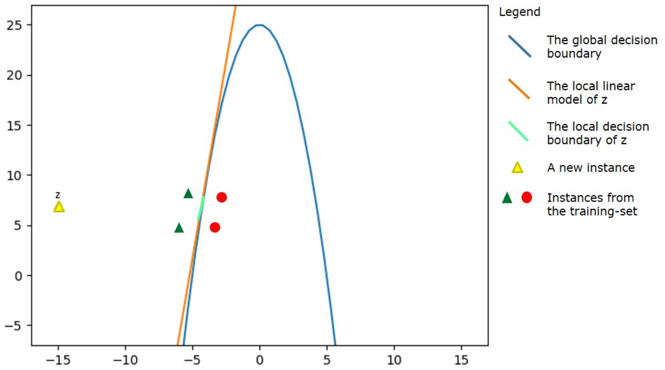


Figure 3.5: The minimum amount of instances needed for a good local linear model is 2 per class in this 2D example. This is the case only if these instances lie along the local decision boundary of $\mathbf{z}$

2. There should be enough instances of both classes. The minimum amount of instances per class that lie along the local decision boundary, which are needed for a good linear approximation is equal to the dimension d of the input space. For example in a two dimensional space, two instances from each class that lie along the local decision boundary of $\mathbf{z}$ are needed as shown in figure 3.5.

3. The instances should be linearly separable according to the classes.

We propose two solutions for sampling of the local training-set of $\mathbf{z}$, taking the the desired characteristics of a local training-set into account. The solutions are illustrated using a two dimensional classification example shown in figure 3.4. The blue parabola is the decision boundary of the black-box ML model. The triangles and circles are instances of different classes from the training-set that the ML model was built on. The local-linear model needs to be computed of a new instance $\mathbf{z}$ (yellow triangle).

The first sampling strategy is called *Boundary Independent Sampling (BIS)*. The main idea is to sample around the closet enemy of $\mathbf{z}$ from the training-set. Its pseudo-code is provided in algorithm 1. Its two steps and motivations are listed below:

1. The local training set $\mathbf{z}$ is sampled around the local boundary of $\mathbf{z}$ (similar to the idea of LS [35]). This makes it possible to sample enough instances from both classes avoiding a bad sampling scenario of LIME (figure 2.6a). We assume that the closest enemy $\mathbf{x}_{border}$ of $\mathbf{z}$ from the training-set lies close to the local decision boundary of $\mathbf{z}$ and sample around $\mathbf{x}_{border}$. The growing spheres

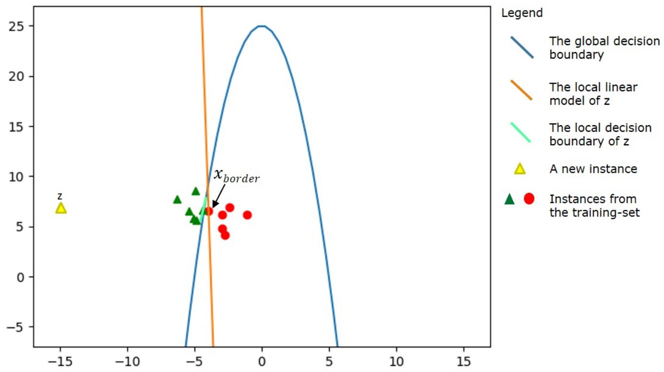


Figure 3.6: Sampling strategy BIS applied onto a two dimensional example.

algorithm (see section 2.2.3) suggested by LS is not used, because it is not accurate in a high dimensional input space.

2. LS had two bad sampling scenarios (figures 2.8a and 2.8c) in which there were too few or too many local training instances, that lead to bad linear approximations. To avoid these scenarios as much as possible, this method proposes to sample $i \cdot d$ examples of each class from the training-set that lie the closest to x_{border}. The number of samples per class is dependent on the input dimension d, because d instances per class are the minimum amount of examples needed for a good linear approximation assuming that these d instances lie along the closest decision boundary of n. These d instances might not lie exactly along the decision boundary, thus the amount is increased with i_{small} which is a small integer greater than 1. This strategy applied on a two dimensional example with $i = 3$ is showed on figure 3.6. The green and red shapes are instances sampled from the training-set (figure 3.4) to build the local linear model of z. The local linear model of z is not completely approximating the local decision boundary correctly. This is because most of the sampled instances do not lie along the local decision boundary of z. The second sampling strategy tries to sample the instances that lie along the decision boundary.

BIS samples training-instances that lie close to x_{border} (figure 3.6), but instances that lie more along the local decision boundary of z and are close to $\mathrm{x}_{b}order$ as shown in figure 3.7e, form a better local training-set. Taking that into account, we propose a second sampling strategy called *Boundary Dependent Sampling (BDS)* that builds upon the first strategy. Its pseudo-code is presented in algorithm 2. Two neighbourhood amounts are defined namely the i_{small} and i_{big}, which correspond to the amount of instances per class and dimension to be sampled form the training-set. i_{small} and i_{big} are respectively a small and a big integer greater than 0, with $i_{small} < i_{big}$. The BDS strategy works as follows:

1. First, a large amount of training instances X_{big} using BIS strategy with $i = i_{big}$ is sampled. Examples of training instances extracted using the BIS strategy with $i_{big} = 10$ and $i_{big} = 50$ on a 2D space, are shown in figure 3.7a and 3.7b, respectively. The main idea is to choose $i_{small} \cdot d$ amount of instances per class from X_{big} that are the most suitable as the local training-set of z.

2. A score is given to each instance of X_{big} which approximates how close this instance lies to the local decision boundary of z. The score of an instance $\mathrm{x}_1 \in X_{big}$ is equal to the smallest euclidean distance from x_1 to an instance of X_{big} that has a different prediction than x_1. Figures 3.7c and 3.7d show the local training-set (with $i_{small} = 3$) of z filtered using this score from $i_{big} = 10$ and $i_{big} = 50$ neighbourhoods, respectively. In the first example, the sampled instances are good to build the local linear model of z, but this is not the case for the second one. In the second case, the big neighbourhood crosses another decision boundary and subsequently training instance from there are included.

3. To avoid the scenario seen as seen in figure 3.7d, the score function is adjusted by taking the euclidean distance from $\mathbf{x}_1$ to $\mathbf{x}_{border}$ into account. This makes sure that the instances close to $\mathbf{x}_{border}$ and to the local decision boundary are preferred over instances that are far away from $\mathbf{x}_{border}$. The final extracted local training-set of $\mathbf{z}$ from $i_{big} = 10$ and $i_{big} = 50$ neighbourhood using this adjusted score are the same and shown in figure 3.7e.

At last, given the local training-set of $\mathbf{z}$, any linear classification algorithm can be used to build the local linear model of $\mathbf{z}$.

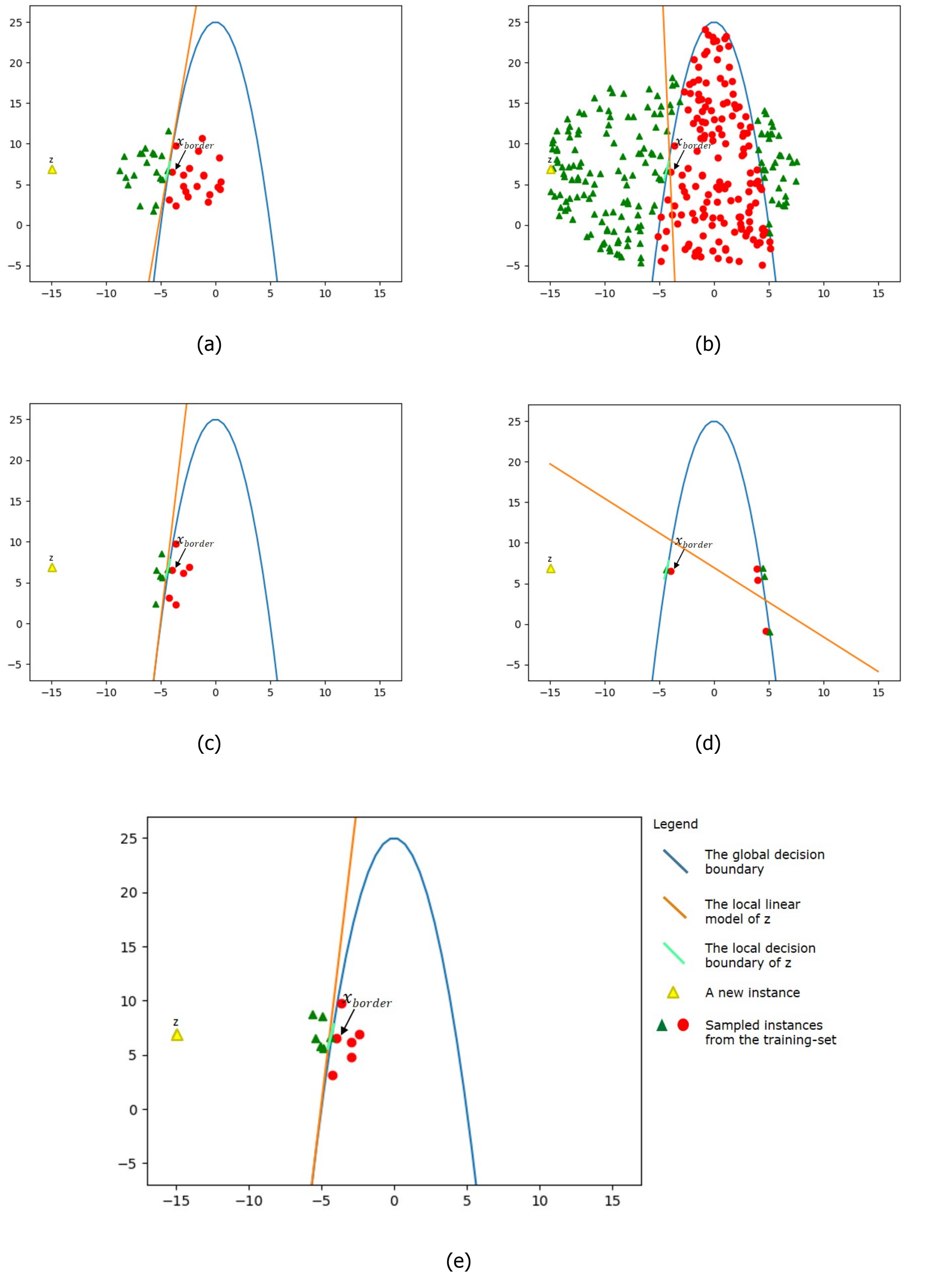


Figure 3.7: These example illustrate the working of the sampling strategy BDS.

```
Algorithm 1: BIS($\mathbf{z}$, $c_z$, $X$, $y$, $i$)
  Input  : $\mathbf{z}$: instance to explain prediction of, $c_z$: predicted label of $\mathbf{z}$, $X$:the training-set, $y$:the
           predicted labels of $\mathbf{z}$, $i_{small}$:the amount of instances per dimension and class to sample
           (default value:3)
  Output: The local training-set of $\mathbf{z}$ ($X_n$) and the corresponding labels $y_n$

  $X_a \leftarrow \{\mathbf{x}_i \in X \mid y_i = c_z\}$          // the allies of n from the training-set
  $X_e \leftarrow \{\mathbf{x}_i \in X \mid y_i \neq c_z\}$          // the enemies of n from the training-set
  $\mathbf{x}_{border} \leftarrow \arg\min \|z - x\|,\ x \in X_e$
  $amount \leftarrow i_{small} \cdot d$          // the amount of instances to sample per class
  $d_a = \{\|\mathbf{x}_{border} - \mathbf{x}\| \mid x \in X_a\}$
  $d_e = \{\|\mathbf{x}_{border} - \mathbf{x}\| \mid x \in X_e\}$
  $X'_a \leftarrow$ sort($X_a$, $key = d_a$)[1 .. $amount$]
  $X'_e \leftarrow$ sort($X_e$, $key = d_e$)[1 .. $amount$]
  $X_z \leftarrow$ concatenate($X'_a$, $X'_e$)
  $y_z \leftarrow$ corresponding $y$ values of $X_z$
  return $X_z$, $y_z$
```

```
Algorithm 2: BDS($\mathbf{z}$, $c_z$, $X$, $y$, $i_{small}$, $i_{big}$)
  Input  : $\mathbf{z}$: instance to explain prediction of, $c_z$: predicted label of $\mathbf{z}$, $X$:the training-set, $y$:the
           predicted labels of X, $i_{small}$:the amount of instances per dimension and class to sample
           (default value: 3), $i_{big}$: the amount of instances per dimension and class to consider
           for sampling (default value: 15)
  Output: The local training-set of $\mathbf{z}$ ($X_n$) and the corresponding labels $y_n$

  $X_{big}, y_{big} \leftarrow$ BIS($n, c_z, X, y, i_{big}$)
  $X_a \leftarrow \{\mathbf{x}_i \in X_{big} \mid y_i = c_z\}$          // the allies of n from $X_{big}$
  $X_e \leftarrow \{\mathbf{x}_i \in X_{big} \mid y_i \neq c_z\}$          // the enemies of n from $X_{big}$
  $x_{border} \leftarrow \arg\min \|\mathbf{z} - \mathbf{x}\|,\ \mathbf{x} \in X_{big}$
  $d_a \leftarrow$ empty list          // distance used to choose final samples
  $d_e \leftarrow$ empty list          // distance used to choose final samples
  for each $\mathbf{u} \in X_{big}$ do
      /* Get the instances in $X_{big}$ that have the opposite class as $\mathbf{u}$          */
      if $\mathbf{u} \in X_a$ then
          $target \leftarrow X_e$
      else
          $target \leftarrow X_a$
      end
      /* Get the shortest distance from $u$ to an instance of $target$          */
      $d_{oc} \leftarrow min(\|\mathbf{u} - \mathbf{x}\|), \mathbf{x} \in target$
      /* Get the distance from $\mathbf{u}$ to $\mathbf{x}_{border}$          */
      $d_b \leftarrow \|\mathbf{x}_{border} - u\|$
      /* Combine the both distances          */
      $final\_distance = d_{oc} + d_b$
      if $u \in X_a$ then
          $d_a.append(final\_distance)$
      else
          $d_e.append(final\_distance)$
      end
  end
  $amount \leftarrow i_{small} \cdot d$          // the amount of instances to sample per class
  $X'_a \leftarrow$ sort($X_a$, $key = d_a$)[1 .. $amount$]
  $X'_e \leftarrow$ sort($X_e$, $key = d_e$)[1 .. $amount$]
  $X_z \leftarrow$ concatenate($X'_a$, $X'_e$)
  $y_z \leftarrow$ corresponding $y_{big}$ values of $X_z$
  return $X_z$, $y_z$
```

3.3. Generating contrastive explanations

One of the important finding from social research is that *why questions* are contrastive (see section 2.1.1). In ML context, if a ML model predicts an instance $\mathbf{z}$ to be of class $c_{\mathbf{z}}$ than the user might ask "Why did the ML model predict this instances as class $c_{\mathbf{z}}$ instead of class c_f". The user expected class c_f and wants a specific explanation of why $c_{\mathbf{z}}$ (the fact class) was predicted instead of c_f (the foil class). If the ML model solves a binary classification, then any explanation is by definition contrastive and the methods describes in the previous sections can be used. On the other hand if the ML model solves a multi-class problem, providing a contrastive explanation is not trivial.

First, the foil class has to be determined. Determining the foil class means understanding which class the user expected instead of the fact class. Performing this task exactly is a very hard task. We propose a heuristic method to get the foil class. Generally, the ML model gives a score for each possible class. A score of a class states how likely the instance $\mathbf{z}$ belongs to that class. The predicted class $c_{\mathbf{z}}$ has the highest score. We regard the class with the second highest score as the foil class c_f.

The instances from the training-set that have the fact or the foil class as predictions are filtered. With this filtered training-set along with the predicted labels, a LEAFAGE explanation can be extracted that explains why the fact class was predicted instead of the foil class.

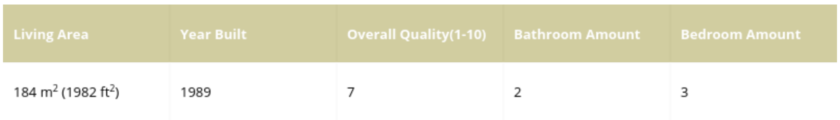
3.4. Explanation extraction

Given the local linear model $g(\mathbf{x}) = w_0 + w_1x_1 + w_2x_2 + ... + w_dx_d$ of an instance $\mathbf{z} = [z_1, .., z_d]$, the importance of each feature for the prediction of $\mathbf{z}$ can be extracted. The prediction of $\mathbf{z}$, can be determined by the score $g(\mathbf{z})$ (see section 2.2.2). Thus, the contribution of a feature value z_i, to the prediction is equal to $abs(w_i * z_i)$. The magnitude of this contribution value denotes the importance of the feature value z_i to the prediction. The importance of each feature value for the prediction can be provided as an explanation to the user.

Social research suggested that explanations are selected (see section 2.1.3) according to different criteria. One of those criterion is responsibility, in which people prefer causes that are more responsible for an event, over others. The importance of each feature value can be seen as how responsible the feature is for the prediction. A small subset of feature values with high importance can be chosen to give as an explanation.

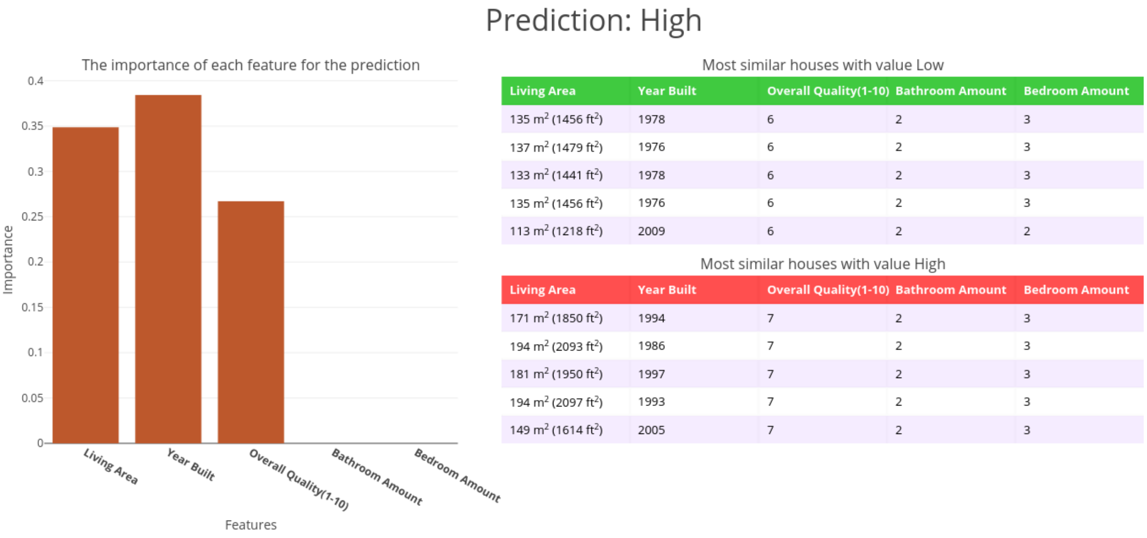
Social research further indicated that Example Based Reasoning lies very close to how we as humans think. It can be used for problem-solving and decision-justification. Given the black-box similarity measure b of $\mathbf{z}$, instances that have similar classification logic can be extracted from the training-set. Furthermore, one can provide similar instances from the fact class and the foil class. That can give insights on the differences between the instances from the fact and foil class in the neighborhood of $\mathbf{z}$.

We propose to combine feature importance-based explanation with example-based explanation. This combination will allow the user to understand the importance of each feature value and the differences between the instances from the fact and foil class. An example of a house that is predicted as value low by a ML model and an LEAFAGE explanation for its prediction are shown in figures 3.8 and 3.9, respectively. The left graph of figure 3.9, shows which of the feature values of the house were the most important to make the prediction. The length of each bar shows the relative importance of each feature. In this case, the feature values *Bathroom Amount=2* and *Bedroom Amount=3* are not important. Feature value Year *Built=1989* is the most important followed by *Living Area=184*m^2 and *Overall Quality=7*. The two tables in the right show houses that are similar to the house being explained. The green table shows 5 similar houses that have low value and the red table shows 5 similar houses with high value. These tables make clear how big the differences for each feature are, between similar houses with value low and high. For example in this explanation, the differences in Living Area are big between low and high value houses while there are no differences in Bathroom Amount and Bedroom Amount.



Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
184 m^2 (1982 ft^2)	1989	7	2	3

Figure 3.8: Example of a house that is predicted as value low by the machine learning model.



Most similar houses with value Low

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
135 m^2 (1456 ft^2)	1978	6	2	3
137 m^2 (1479 ft^2)	1976	6	2	3
133 m^2 (1441 ft^2)	1978	6	2	3
135 m^2 (1456 ft^2)	1976	6	2	3
113 m^2 (1218 ft^2)	2009	6	2	2

Most similar houses with value High

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
171 m^2 (1850 ft^2)	1994	7	2	3
194 m^2 (2093 ft^2)	1986	7	2	3
181 m^2 (1950 ft^2)	1997	7	2	3
194 m^2 (2097 ft^2)	1993	7	2	3
149 m^2 (1614 ft^2)	2005	7	2	3

Figure 3.9: Example of a LEAFAGE explanation.

4

Quantitative Evaluation

This chapter evaluates the ability of LEAFAGE in reflecting the true local reasoning of the underlying black-box ML model. This subsection demonstrates the model-agnostic character of LEAFAGE by evaluating quantitatively how faithful the extracted explanations are to six different types of black-box ML models applied on four different UCI[1] datasets and one artificial dataset. It compares the fidelity of LEAFAGE's BIS and BDS strategy, LIME and a baseline model, with each other.

4.1. Evaluation method

A dataset is first split into two disjoint sets of training-set (X_{train}, y_{train}) and testing-set (X_{test}, y_{test}), which contain 70% and 30% of data, respectively. The training-set is used to build a black-box classifier f. Further, a local linear model is built for each instance of the testing-set. To build these local linear models, only the training-set and the black-box classifier f are available. The testing instances X_{test} with their predicted labels $y_{test\_f} = \{f(\mathrm{x}_i) | \mathrm{x}_i \in X_{test}\}$ are used to get a *local fidelity score* of each local linear model. At last, the individual local fidelity score of all the local linear models are averaged to get one fidelity score per dataset and classifier. We propose a new method of getting the local fidelity score of a local linear model $g_{\mathbf{z}}$ of an instance z with $f(\mathrm{z}) = c_{\mathbf{z}}$

The local linear model should be valid in the neighbourhood of $\mathbf{z}$. The difficulty lies in how to set the size of this neighbourhood. Method LS [35] suggested to test the performance of $g_{\mathbf{z}}$ on the testing instances that fall into a hyper-sphere with a fixed radius and $\mathbf{z}$ as center. Having fixed radius can lead to too small or too big neighbourhoods (see section 2.1.4). We propose a custom radius per $g_{\mathbf{z}}$. The hyper-sphere with center $\mathbf{z}$ is expanded until it includes p percentage of instances that do not have the same predicted label as $c_{\mathbf{z}}$. p should be smaller than and close to one (p = 0.95 is used in the experiments), such that the closest testing instances of the opposite class of z are included and to make the evaluation local, respectively. The test instances $X_{test\_\mathbf{z}}$ that fall into this hyper-sphere are used to get the fidelity score of $g_{\mathbf{z}}$. The labels given by the black-box classifier f are compared to the scores given by the local linear model $g_{\mathbf{z}}$, using the AUC evaluation metric. The local fidelity score of a local linear model $g_{\mathbf{z}}$ of instance z, to a black-box classifier f is defined as follows:

$$AUC(\{f(\mathrm{x}_i) | \mathrm{x}_i \in X_{test\_\mathbf{z}}\}, \{g_{\mathbf{z}}(\mathrm{x}_i) | \mathrm{x}_i \in X_{test\_\mathbf{z}}\})$$

The average local fidelity scores of four strategies are compared per dataset and classifier. The first two strategies are *Boundary Independent Sampling* (BIS) and *Boundary Dependent Sampling* (BDS) of LEAFAGE (see section 3.2). The third strategy is the current state-of-art model-agnostic method that extracts local linear models called LIME (see section 2.2.3). At last, a baseline method is used, that always predicts the class given by f, of the instance being explained.

[1] Hyperlink UCI datasets

4.2. Datasets and black-box models

The different black-box ML models used are namely *Logistic Regression*[2] (LR), *Support Vector Machine* with linear kernel[3] (SVM), *Linear Discriminant Analysis*[4] (LDA), *Random Forest*[5] (RF), *Decision Tree*[6] (DT) and *K Nearest Neighbour*[7] with $K = 1$ (KNN). *Scikit-learn 0.19.2* [8] library was used to build these models with its default hyper-parameter unless stated otherwise.

These classifiers are applied to 5 different datasets with different number of features, rows and complexities (see table 4.1) and described below. Multi-class datasets with n classes are converted into n binary datasets of one-vs-rest fashion. We call a combination of a binary dataset and a classifier *a setting*. In total there are 54 settings.

Dataset	Number of features	Number of rows	Number of classes	Amount per class
Iris	4	150	3	50/50/50
Wine	13	178	3	59/71/48
Breast Cancer	32	569	2	212/357
Bank Note	4	1372	2	762/610
AD	2	500	2	250/250

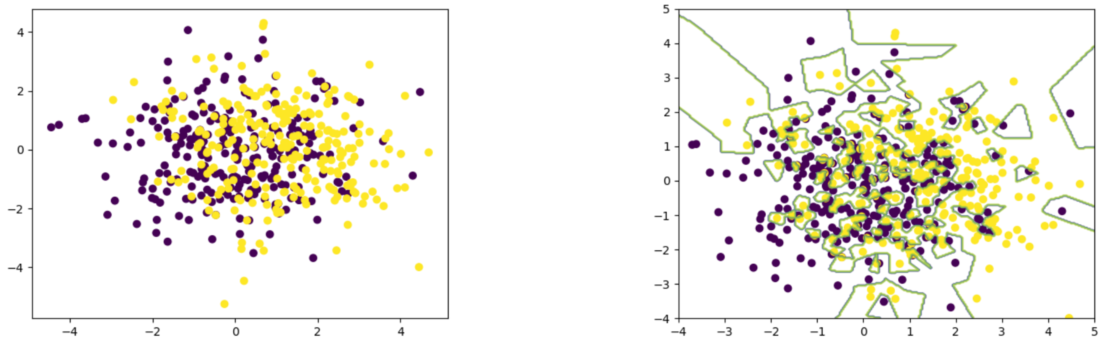
Table 4.1: UCI datasets used for quantitative evaluation

- *Iris*: This is a famous classification task of predicting the type of iris flower given it's different attributes about the petal and sepal [41].
- *Wine*: This dataset contains the chemical analysis (independent variables) of three different types of Italian wines (dependent variable) [42].
- *Breast Cancer*: It is a binary classification task of classifying whether a breast mass is benign or malignant given different features about it [43].
- *Banknote*: This a binary classification problem of predicting the authenticity of a banknote as *fake* or *real*, given its different numerical characteristics [44].
- *Artificial dataset*: At last, we include a two-dimensional binary artificial dataset (AD) with highly non-separable classes (figure 4.1a). The instances from the purple and yellow classes are samples from bi-variate normal distribution with different means ($[0, 0]$ and $[0, 1]$, respectively) and the same covariance matrix ($\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$). The KNN classifier trained on this dataset is visualized on figure 4.1b. The decision boundary is complex and highly non-linear.

4.3. Results

Per setting, the average fidelity score along with the standard deviation in brackets is presented on figure 4.2. LEAFAGE strategies are computed with $i_{small} = 10$. Further, we perform statistical tests per setting between the strategy with the highest mean and the rest of the strategies. The strategy with the highest mean along with the other strategies that are not significantly different are denoted with a dark font color. While the rest of the strategies are denoted with a light font color. We perform a Wilcoxon signed-ranked test which tests the null hypothesis that two related paired samples come form the same distribution. The statistical tests are performed with a critical value of 0.05 with Bonferroni correction.

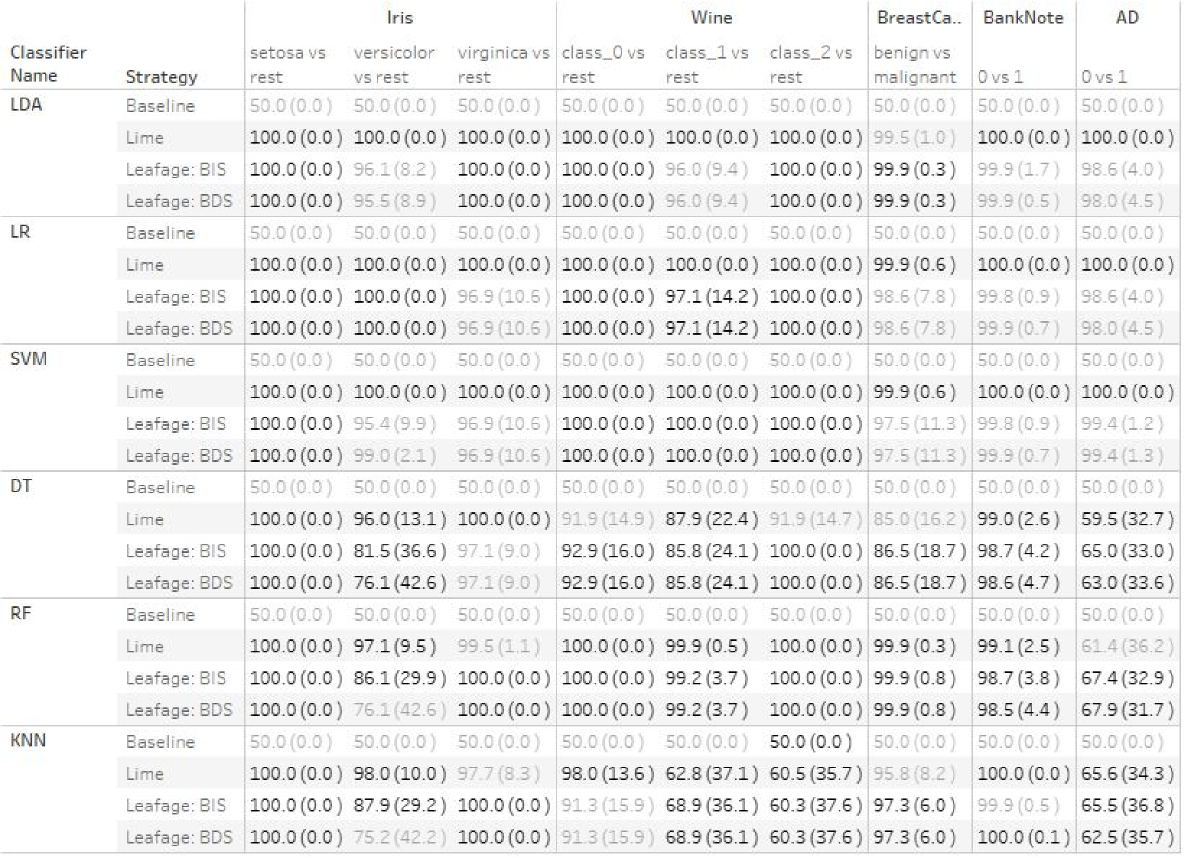
[2] Hyperlink Scikit-learn's Logictic Regression
[3] Hyperlink Scikit-learn's Support Vector Machine
[4] Hyperlink Scikit-learn's Linear Discriminant Analysis
[5] Hyperlink Scikit-learn's Random Forest
[6] Hyperlink Scikit-learn's Decision Tree
[7] Hyperlink Scikit-learn's K Nearest Neighbours
[8] Hyperlink Scikit-learn



(a) A highly non-linearly separable binary dataset.

(b) A KNN model trained on the *AD* dataset.

Figure 4.1



		Iris			Wine			BreastCa..	BankNote	AD
Classifier Name	Strategy	setosa vs rest	versicolor vs rest	virginica vs rest	class_0 vs rest	class_1 vs rest	class_2 vs rest	benign vs malignant	0 vs 1	0 vs 1
LDA	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
	Lime	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	99.5 (1.0)	100.0 (0.0)	100.0 (0.0)
	Leafage: BIS	100.0 (0.0)	96.1 (8.2)	100.0 (0.0)	100.0 (0.0)	96.0 (9.4)	100.0 (0.0)	99.9 (0.3)	99.9 (1.7)	98.6 (4.0)
	Leafage: BDS	100.0 (0.0)	95.5 (8.9)	100.0 (0.0)	100.0 (0.0)	96.0 (9.4)	100.0 (0.0)	99.9 (0.3)	99.9 (0.5)	98.0 (4.5)
LR	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
	Lime	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	99.9 (0.6)	100.0 (0.0)	100.0 (0.0)
	Leafage: BIS	100.0 (0.0)	100.0 (0.0)	96.9 (10.6)	100.0 (0.0)	97.1 (14.2)	100.0 (0.0)	98.6 (7.8)	99.8 (0.9)	98.6 (4.0)
	Leafage: BDS	100.0 (0.0)	100.0 (0.0)	96.9 (10.6)	100.0 (0.0)	97.1 (14.2)	100.0 (0.0)	98.6 (7.8)	99.9 (0.7)	98.0 (4.5)
SVM	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
	Lime	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	99.9 (0.6)	100.0 (0.0)	100.0 (0.0)
	Leafage: BIS	100.0 (0.0)	95.4 (9.9)	96.9 (10.6)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	97.5 (11.3)	99.8 (0.9)	99.4 (1.2)
	Leafage: BDS	100.0 (0.0)	99.0 (2.1)	96.9 (10.6)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	97.5 (11.3)	99.9 (0.7)	99.4 (1.3)
DT	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
	Lime	100.0 (0.0)	96.0 (13.1)	100.0 (0.0)	91.9 (14.9)	87.9 (22.4)	91.9 (14.7)	85.0 (16.2)	99.0 (2.6)	59.5 (32.7)
	Leafage: BIS	100.0 (0.0)	81.5 (36.6)	97.1 (9.0)	92.9 (16.0)	85.8 (24.1)	100.0 (0.0)	86.5 (18.7)	98.7 (4.2)	65.0 (33.0)
	Leafage: BDS	100.0 (0.0)	76.1 (42.6)	97.1 (9.0)	92.9 (16.0)	85.8 (24.1)	100.0 (0.0)	86.5 (18.7)	98.6 (4.7)	63.0 (33.6)
RF	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
	Lime	100.0 (0.0)	97.1 (9.5)	99.5 (1.1)	100.0 (0.0)	99.9 (0.5)	100.0 (0.0)	99.9 (0.3)	99.1 (2.5)	61.4 (36.2)
	Leafage: BIS	100.0 (0.0)	86.1 (29.9)	100.0 (0.0)	100.0 (0.0)	99.2 (3.7)	100.0 (0.0)	99.9 (0.8)	98.7 (3.8)	67.4 (32.9)
	Leafage: BDS	100.0 (0.0)	76.1 (42.6)	100.0 (0.0)	100.0 (0.0)	99.2 (3.7)	100.0 (0.0)	99.9 (0.8)	98.5 (4.4)	67.9 (31.7)
KNN	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
	Lime	100.0 (0.0)	98.0 (10.0)	97.7 (8.3)	98.0 (13.6)	62.8 (37.1)	60.5 (35.7)	95.8 (8.2)	100.0 (0.0)	65.6 (34.3)
	Leafage: BIS	100.0 (0.0)	87.9 (29.2)	100.0 (0.0)	91.3 (15.9)	68.9 (36.1)	60.3 (37.6)	97.3 (6.0)	99.9 (0.5)	65.5 (36.8)
	Leafage: BDS	100.0 (0.0)	75.2 (42.2)	100.0 (0.0)	91.3 (15.9)	68.9 (36.1)	60.3 (37.6)	97.3 (6.0)	100.0 (0.1)	62.5 (35.7)

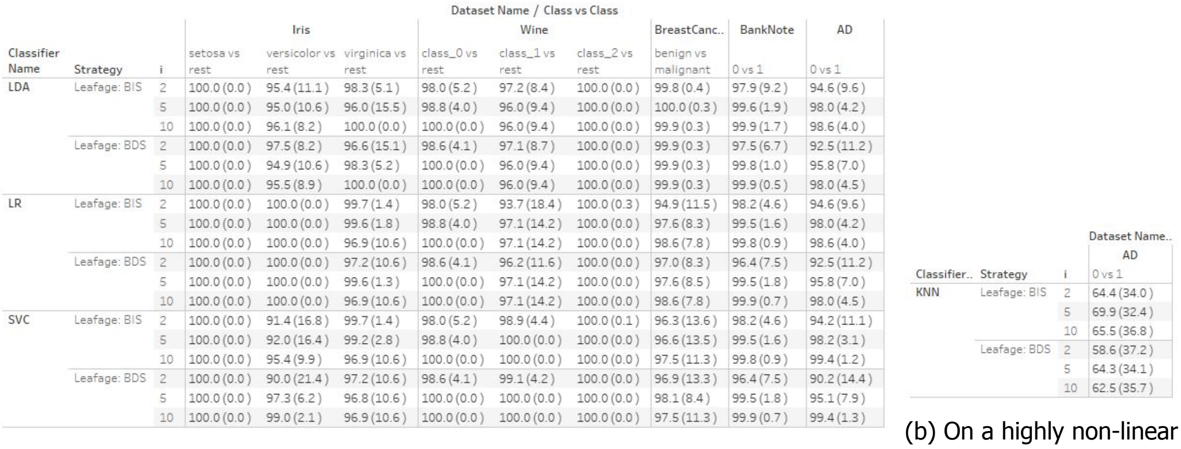
Figure 4.2: The average local fidelity per setting with the standard deviation in brackets. The strategy with the highest mean along with other strategies that are not significantly different are denoted with a dark font color.

First, we view the performance of LIME, BIS and BDS versus the baseline model. All three methods have a consistently average score higher than the baseline model over all settings.

Further, we establish whether there is a difference in performance between ML models with linear (SVM, LDA, LR) and non-linear (DT, RF, KNN) decision boundaries. We expect that the strategies perform better on the former models in comparison to the latter, because BIS, BDS and LIME are based on linear models. Indeed, all three strategies do perform better on the former models. This is very clear with the highly linearly non-separable *AD* dataset. On the models with linear decision boundaries these strategies have an average fidelity score of greater than 98%, while on non-linear models the scores are in the sixties.

Next, we look at the differences between LIME and LEAFAGE BIS and BDS. Overall none of the strategies are consistently better than the others. On the models with linear decision boundary LIME scores significantly better than the others in 13 out of 27 setting. While on the non-linear models LEAFAGE strategies score significantly better than LIME on 7 out of 27 settings. The better performance of LIME on models with linear decision boundary could be because LIME uses a high amount of samples over the whole input space to fit the local linear model. The strategies of LEAFAGE on the other hand samples around the closest decision boundary and limits the sampling amount to a minimum. This might explain the better performance of LEAFAGE over LIME on non-linear models. No significant difference between LEAFAGE BIS and BDS is present.

Furthermore, we look at the influence of varying i_{small} on the performance of LEAFAGE BIS and BDS. Figures 4.3a and 4.3b show the results while varying i_{small} on models with linear (SVM, LDA, LR) and highly non-linear (KNN) decision boundaries, respectively. Generally on models with linear decision boundaries, the performance of both strategies increases with i_{small}. This is expected because increasing the sampling neighbourhood will allow to approximate the global linear decision boundary better. On the other hand if the outcome of the black-box classifier is highly non-linear a smaller value for i_{small} is better. This can be seen on the KNN model i.e. the performance decreases from $i_{small} = 5$ to $i_{small} = 10$.



			Dataset Name / Class vs Class							
			Iris			Wine			BreastCanc..	BankNote	AD
Classifier Name	Strategy	i	setosa vs rest	versicolor vs rest	virginica vs rest	class_0 vs rest	class_1 vs rest	class_2 vs rest	benign vs malignant	0 vs 1	0 vs 1
LDA	Leafage: BIS	2	100.0 (0.0)	95.4 (11.1)	98.3 (5.1)	98.0 (5.2)	97.2 (8.4)	100.0 (0.0)	99.8 (0.4)	97.9 (9.2)	94.6 (9.6)
		5	100.0 (0.0)	95.0 (10.6)	96.0 (15.5)	98.8 (4.0)	96.0 (9.4)	100.0 (0.0)	100.0 (0.3)	99.6 (1.9)	98.0 (4.2)
		10	100.0 (0.0)	96.1 (8.2)	100.0 (0.0)	100.0 (0.0)	96.0 (9.4)	100.0 (0.0)	99.9 (0.3)	99.9 (1.7)	98.6 (4.0)
	Leafage: BDS	2	100.0 (0.0)	97.5 (8.2)	96.6 (15.1)	98.6 (4.1)	97.1 (8.7)	100.0 (0.0)	99.9 (0.3)	97.5 (6.7)	92.5 (11.2)
		5	100.0 (0.0)	94.9 (10.6)	98.3 (5.2)	100.0 (0.0)	96.0 (9.4)	100.0 (0.0)	99.9 (0.3)	99.8 (1.0)	95.8 (7.0)
		10	100.0 (0.0)	95.5 (8.9)	100.0 (0.0)	100.0 (0.0)	96.0 (9.4)	100.0 (0.0)	99.9 (0.3)	99.9 (0.5)	98.0 (4.5)
LR	Leafage: BIS	2	100.0 (0.0)	100.0 (0.0)	99.7 (1.4)	98.0 (5.2)	93.7 (18.4)	100.0 (0.3)	94.9 (11.5)	98.2 (4.6)	94.6 (9.6)
		5	100.0 (0.0)	100.0 (0.0)	99.6 (1.8)	98.8 (4.0)	97.1 (14.2)	100.0 (0.0)	97.6 (8.3)	99.5 (1.6)	98.0 (4.2)
		10	100.0 (0.0)	100.0 (0.0)	96.9 (10.6)	100.0 (0.0)	97.1 (14.2)	100.0 (0.0)	98.6 (7.8)	99.8 (0.9)	98.6 (4.0)
	Leafage: BDS	2	100.0 (0.0)	100.0 (0.0)	97.2 (10.6)	98.6 (4.1)	96.2 (11.6)	100.0 (0.0)	97.0 (8.3)	96.4 (7.5)	92.5 (11.2)
		5	100.0 (0.0)	100.0 (0.0)	99.6 (1.3)	100.0 (0.0)	97.1 (14.2)	100.0 (0.0)	97.6 (8.5)	99.5 (1.8)	95.8 (7.0)
		10	100.0 (0.0)	100.0 (0.0)	96.9 (10.6)	100.0 (0.0)	97.1 (14.2)	100.0 (0.0)	98.6 (7.8)	99.9 (0.7)	98.0 (4.5)
SVC	Leafage: BIS	2	100.0 (0.0)	91.4 (16.8)	99.7 (1.4)	98.0 (5.2)	98.9 (4.4)	100.0 (0.1)	96.3 (13.6)	98.2 (4.6)	94.2 (11.1)
		5	100.0 (0.0)	92.0 (16.4)	99.2 (2.8)	98.8 (4.0)	100.0 (0.0)	100.0 (0.0)	96.6 (13.5)	99.5 (1.6)	98.2 (3.1)
		10	100.0 (0.0)	95.4 (9.9)	96.9 (10.6)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	97.5 (11.3)	99.8 (0.9)	99.4 (1.2)
	Leafage: BDS	2	100.0 (0.0)	90.0 (21.4)	97.2 (10.6)	98.6 (4.1)	99.1 (4.2)	100.0 (0.0)	96.9 (13.3)	96.4 (7.5)	90.2 (14.4)
		5	100.0 (0.0)	97.3 (6.2)	96.8 (10.6)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	98.1 (8.4)	99.5 (1.8)	95.1 (7.9)
		10	100.0 (0.0)	99.0 (2.1)	96.9 (10.6)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	97.5 (11.3)	99.9 (0.7)	99.4 (1.3)

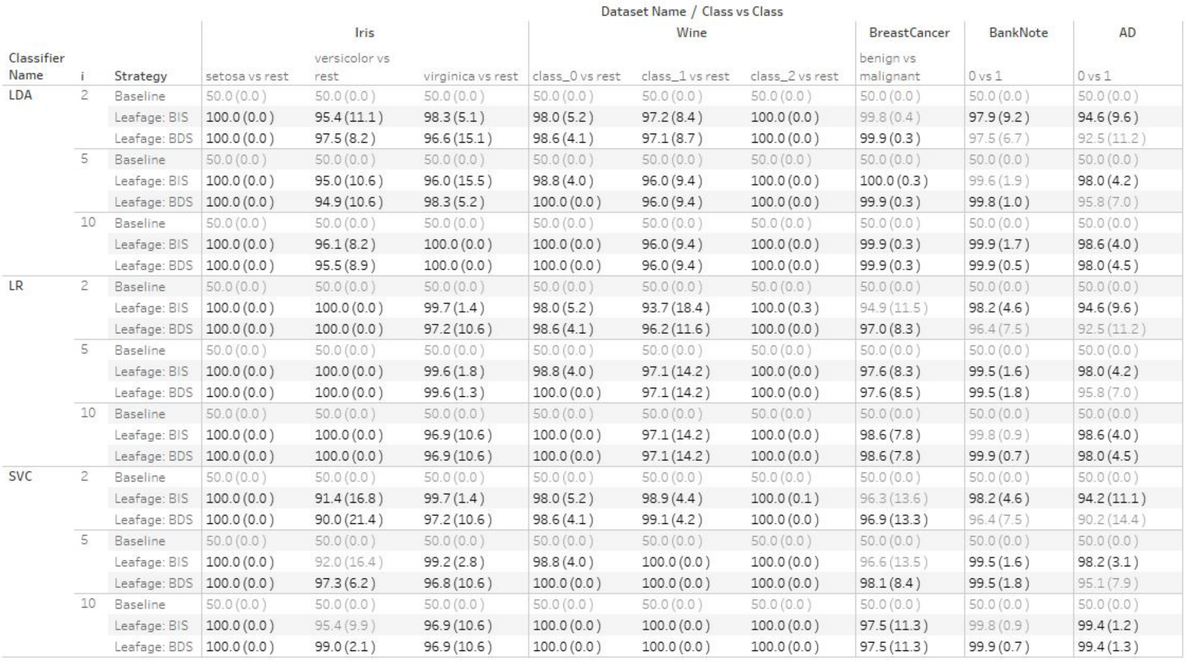
(a) On models with linear decision boundary

			Dataset Name..
			AD
Classifier..	Strategy	i	0 vs 1
KNN	Leafage: BIS	2	64.4 (34.0)
		5	69.9 (32.4)
		10	65.5 (36.8)
	Leafage: BDS	2	58.6 (37.2)
		5	64.3 (34.1)
		10	62.5 (35.7)

(b) On a highly non-linear ML model

Figure 4.3: Fidelity results of LEAFAGE strategies while varying i_{small}.

At last, we establish the differences between LEAFAGE BIS and BDS. Applied on models with linear decision boundary (figure 4.4), BDS performs significantly better than BIS on a high dimensional dataset with a small value for i_{small} (dataset *Breast Cancer* with dimension 32 and $i_{small} = 2$). While on non-linear models (figure 4.5), BIS performs significantly better than BDS on 15 out of 81 settings. This better performance, might be because BDS samples along the non-linear decision boundary while BIS around the closest enemy. BIS samples will include a smaller region of the decision boundary than BDS. A smaller region is more likely to be less non-linear, which leads to a better linear approximation

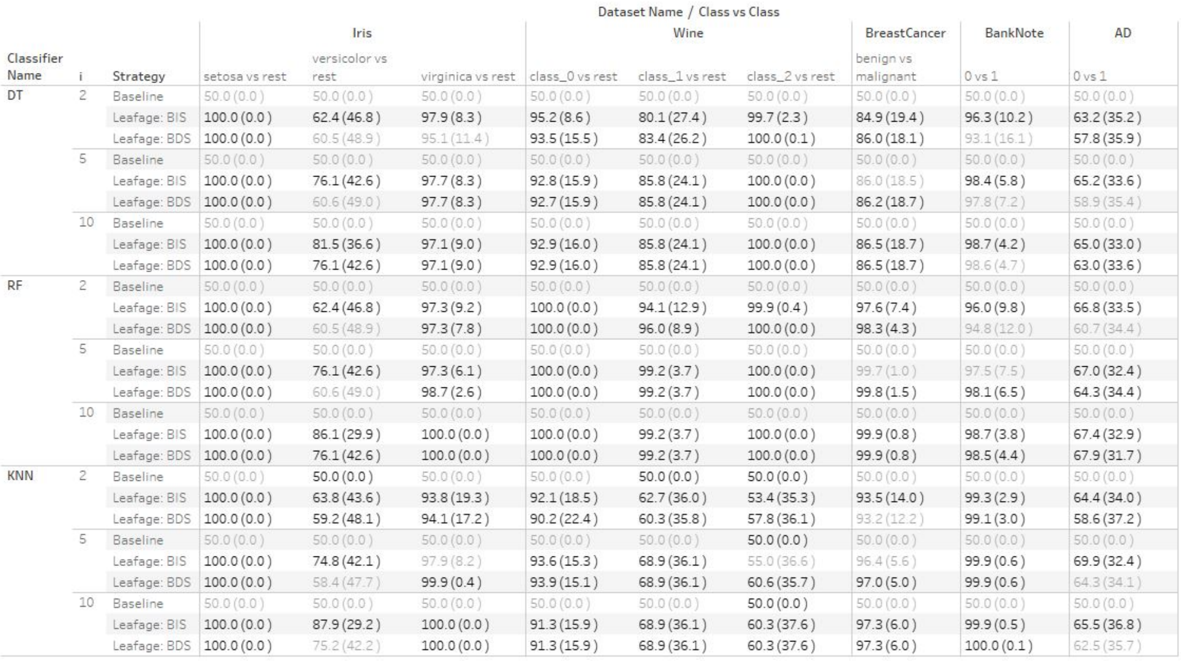


			Dataset Name / Class vs Class								
			Iris			Wine			BreastCancer	BankNote	AD
Classifier Name	i	Strategy	setosa vs rest	versicolor vs rest	virginica vs rest	class_0 vs rest	class_1 vs rest	class_2 vs rest	benign vs malignant	0 vs 1	0 vs 1
LDA	2	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
		Leafage: BIS	100.0 (0.0)	95.4 (11.1)	98.3 (5.1)	98.0 (5.2)	97.2 (8.4)	100.0 (0.0)	99.8 (0.4)	97.9 (9.2)	94.6 (9.6)
		Leafage: BDS	100.0 (0.0)	97.5 (8.2)	96.6 (15.1)	98.6 (4.1)	97.1 (8.7)	100.0 (0.0)	99.9 (0.3)	97.5 (6.7)	92.5 (11.2)
	5	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
		Leafage: BIS	100.0 (0.0)	95.0 (10.6)	96.0 (15.5)	98.8 (4.0)	96.0 (9.4)	100.0 (0.0)	100.0 (0.3)	99.6 (1.9)	98.0 (4.2)
		Leafage: BDS	100.0 (0.0)	94.9 (10.6)	98.3 (5.2)	100.0 (0.0)	96.0 (9.4)	100.0 (0.0)	99.9 (0.3)	99.8 (1.0)	95.8 (7.0)
	10	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
		Leafage: BIS	100.0 (0.0)	96.1 (8.2)	100.0 (0.0)	100.0 (0.0)	96.0 (9.4)	100.0 (0.0)	99.9 (0.3)	99.9 (1.7)	98.6 (4.0)
		Leafage: BDS	100.0 (0.0)	95.5 (8.9)	100.0 (0.0)	100.0 (0.0)	96.0 (9.4)	100.0 (0.0)	99.9 (0.3)	99.9 (0.5)	98.0 (4.5)
LR	2	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
		Leafage: BIS	100.0 (0.0)	100.0 (0.0)	99.7 (1.4)	98.0 (5.2)	93.7 (18.4)	100.0 (0.3)	94.9 (11.5)	98.2 (4.6)	94.6 (9.6)
		Leafage: BDS	100.0 (0.0)	100.0 (0.0)	97.2 (10.6)	98.6 (4.1)	96.2 (11.6)	100.0 (0.0)	97.0 (8.3)	96.4 (7.5)	92.5 (11.2)
	5	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
		Leafage: BIS	100.0 (0.0)	100.0 (0.0)	99.6 (1.8)	98.8 (4.0)	97.1 (14.2)	100.0 (0.0)	97.6 (8.3)	99.5 (1.6)	98.0 (4.2)
		Leafage: BDS	100.0 (0.0)	100.0 (0.0)	99.6 (1.3)	100.0 (0.0)	97.1 (14.2)	100.0 (0.0)	97.6 (8.5)	99.5 (1.8)	95.8 (7.0)
	10	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
		Leafage: BIS	100.0 (0.0)	100.0 (0.0)	96.9 (10.6)	100.0 (0.0)	97.1 (14.2)	100.0 (0.0)	98.6 (7.8)	99.8 (0.9)	98.6 (4.0)
		Leafage: BDS	100.0 (0.0)	100.0 (0.0)	96.9 (10.6)	100.0 (0.0)	97.1 (14.2)	100.0 (0.0)	98.6 (7.8)	99.9 (0.7)	98.0 (4.5)
SVC	2	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
		Leafage: BIS	100.0 (0.0)	91.4 (16.8)	99.7 (1.4)	98.0 (5.2)	98.9 (4.4)	100.0 (0.1)	96.3 (13.6)	98.2 (4.6)	94.2 (11.1)
		Leafage: BDS	100.0 (0.0)	90.0 (21.4)	97.2 (10.6)	98.6 (4.1)	99.1 (4.2)	100.0 (0.0)	96.9 (13.3)	96.4 (7.5)	90.2 (14.4)
	5	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
		Leafage: BIS	100.0 (0.0)	92.0 (16.4)	99.2 (2.8)	98.8 (4.0)	100.0 (0.0)	100.0 (0.0)	96.6 (13.5)	99.5 (1.6)	98.2 (3.1)
		Leafage: BDS	100.0 (0.0)	97.3 (6.2)	96.8 (10.6)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	98.1 (8.4)	99.5 (1.8)	95.1 (7.9)
	10	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
		Leafage: BIS	100.0 (0.0)	95.4 (9.9)	96.9 (10.6)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	97.5 (11.3)	99.8 (0.9)	99.4 (1.2)
		Leafage: BDS	100.0 (0.0)	99.0 (2.1)	96.9 (10.6)	100.0 (0.0)	100.0 (0.0)	100.0 (0.0)	97.5 (11.3)	99.9 (0.7)	99.4 (1.3)

Figure 4.4: The average local fidelity per setting (models with linear decision boundaries) with the standard deviation in brackets of LEAFAGE Strategies. The strategy with the highest mean along with other strategies that are not significantly different are denoted with a dark font color.

of closest decision boundary.

In conclusion, there is no clear winner among the different methods. LIME, BIS and BDS perform consistently better than the baseline model. Overall LIME performs better than the other strategies, on model with linear decision boundaries. However, BIS and BDS perform better than LIME on non-linear models. Between BIS and BDS, BIS performs slightly better on non-linear models.



			Dataset Name / Class vs Class								
			Iris			Wine			BreastCancer	BankNote	AD
Classifier Name	i	Strategy	setosa vs rest	versicolor vs rest	virginica vs rest	class_0 vs rest	class_1 vs rest	class_2 vs rest	benign vs malignant	0 vs 1	0 vs 1
DT	2	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
		Leafage: BIS	100.0 (0.0)	62.4 (46.8)	97.9 (8.3)	95.2 (8.6)	80.1 (27.4)	99.7 (2.3)	84.9 (19.4)	96.3 (10.2)	63.2 (35.2)
		Leafage: BDS	100.0 (0.0)	60.5 (48.9)	95.1 (11.4)	93.5 (15.5)	83.4 (26.2)	100.0 (0.1)	86.0 (18.1)	93.1 (16.1)	57.8 (35.9)
	5	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
		Leafage: BIS	100.0 (0.0)	76.1 (42.6)	97.7 (8.3)	92.8 (15.9)	85.8 (24.1)	100.0 (0.0)	86.0 (18.5)	98.4 (5.8)	65.2 (33.6)
		Leafage: BDS	100.0 (0.0)	60.6 (49.0)	97.7 (8.3)	92.7 (15.9)	85.8 (24.1)	100.0 (0.0)	86.2 (18.7)	97.8 (7.2)	58.9 (35.4)
	10	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
		Leafage: BIS	100.0 (0.0)	81.5 (36.6)	97.1 (9.0)	92.9 (16.0)	85.8 (24.1)	100.0 (0.0)	86.5 (18.7)	98.7 (4.2)	65.0 (33.0)
		Leafage: BDS	100.0 (0.0)	76.1 (42.6)	97.1 (9.0)	92.9 (16.0)	85.8 (24.1)	100.0 (0.0)	86.5 (18.7)	98.6 (4.7)	63.0 (33.6)
RF	2	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
		Leafage: BIS	100.0 (0.0)	62.4 (46.8)	97.3 (9.2)	100.0 (0.0)	94.1 (12.9)	99.9 (0.4)	97.6 (7.4)	96.0 (9.8)	66.8 (33.5)
		Leafage: BDS	100.0 (0.0)	60.5 (48.9)	97.3 (7.8)	100.0 (0.0)	96.0 (8.9)	100.0 (0.0)	98.3 (4.3)	94.8 (12.0)	60.7 (34.4)
	5	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
		Leafage: BIS	100.0 (0.0)	76.1 (42.6)	97.3 (6.1)	100.0 (0.0)	99.2 (3.7)	100.0 (0.0)	99.7 (1.0)	97.5 (7.5)	67.0 (32.4)
		Leafage: BDS	100.0 (0.0)	60.6 (49.0)	98.7 (2.6)	100.0 (0.0)	99.2 (3.7)	100.0 (0.0)	99.8 (1.5)	98.1 (6.5)	64.3 (34.4)
	10	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
		Leafage: BIS	100.0 (0.0)	86.1 (29.9)	100.0 (0.0)	100.0 (0.0)	99.2 (3.7)	100.0 (0.0)	99.9 (0.8)	98.7 (3.8)	67.4 (32.9)
		Leafage: BDS	100.0 (0.0)	76.1 (42.6)	100.0 (0.0)	100.0 (0.0)	99.2 (3.7)	100.0 (0.0)	99.9 (0.8)	98.5 (4.4)	67.9 (31.7)
KNN	2	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
		Leafage: BIS	100.0 (0.0)	63.8 (43.6)	93.8 (19.3)	92.1 (18.5)	62.7 (36.0)	53.4 (35.3)	93.5 (14.0)	99.3 (2.9)	64.4 (34.0)
		Leafage: BDS	100.0 (0.0)	59.2 (48.1)	94.1 (17.2)	90.2 (22.4)	60.3 (35.8)	57.8 (36.1)	93.2 (12.2)	99.1 (3.0)	58.6 (37.2)
	5	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
		Leafage: BIS	100.0 (0.0)	74.8 (42.1)	97.9 (8.2)	93.6 (15.3)	68.9 (36.1)	55.0 (36.6)	96.4 (5.6)	99.9 (0.6)	69.9 (32.4)
		Leafage: BDS	100.0 (0.0)	58.4 (47.7)	99.9 (0.4)	93.9 (15.1)	68.9 (36.1)	60.6 (35.7)	97.0 (5.0)	99.9 (0.6)	64.3 (34.1)
	10	Baseline	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)	50.0 (0.0)
		Leafage: BIS	100.0 (0.0)	87.9 (29.2)	100.0 (0.0)	91.3 (15.9)	68.9 (36.1)	60.3 (37.6)	97.3 (6.0)	99.9 (0.5)	65.5 (36.8)
		Leafage: BDS	100.0 (0.0)	75.2 (42.2)	100.0 (0.0)	91.3 (15.9)	68.9 (36.1)	60.3 (37.6)	97.3 (6.0)	100.0 (0.1)	62.5 (35.7)

Figure 4.5: The average local fidelity per setting (models with non linear decision boundaries) with the standard deviation in brackets of LEAFAGE Strategies. The strategy with the highest mean along with other strategies that are not significantly different are denoted with a dark font color.

5 Empirical Evaluation

To assess the real usefulness of LEAFAGE we perform a user-study. This user-study researches how useful the different parts of a LEAFAGE explanation are to the user. More specifically we evaluate the usefulness of example-based and feature importance-based explanations in terms of the perceived aid in decision making, acceptance and measured transparency.

Each participant is introduced to the user-study as follows:

> Imagine you are looking to buy a property. You are searching online and you find a couple of houses you like. Because it is a big investment, you are interested in the real value of the house. You find a smart application called LEAFAGE that estimates the value of a house automatically given its different features. For simplicity, let's say it predicts whether a property has low or high value. LEAFAGE provides four types of explanation, indicating why the house was predicted with a certain value. In this study we want to understand which type of these explanations are useful to the user in decision making.

5.1. Dataset used

The user-study will be applied on the IOWA housing dataset [45]. It is an online free available public dataset that describes the sale of individual residential property in Ames, Iowa from 2006 to 2010. It has been made available by the Ames City Assessor's Office in 2011. Different important characteristics along with the sale price per property are specified in the dataset.

We consider houses with sale price lower than 150.000$ and higher than 200.000$ as value *low* and *high*, respectively. Houses in-between those prices are removed, such that an ML model can be built that can predict the sale value (low or high) with high performance. A bad reasoning of the underlying ML model can affect the perception of an explanation negatively. To mitigate this influence, we choose a high performing ML model. The resulting dataset contains 619 houses with value low and 427 houses with value high. Furthermore, we use 5 features of each of these houses to estimate their value. These features are *Living Area*, *Year Built*, *Overall Quality*, *Bathroom Amount* and *Bedroom Amount*. They are chosen because they are understandable to the general public.

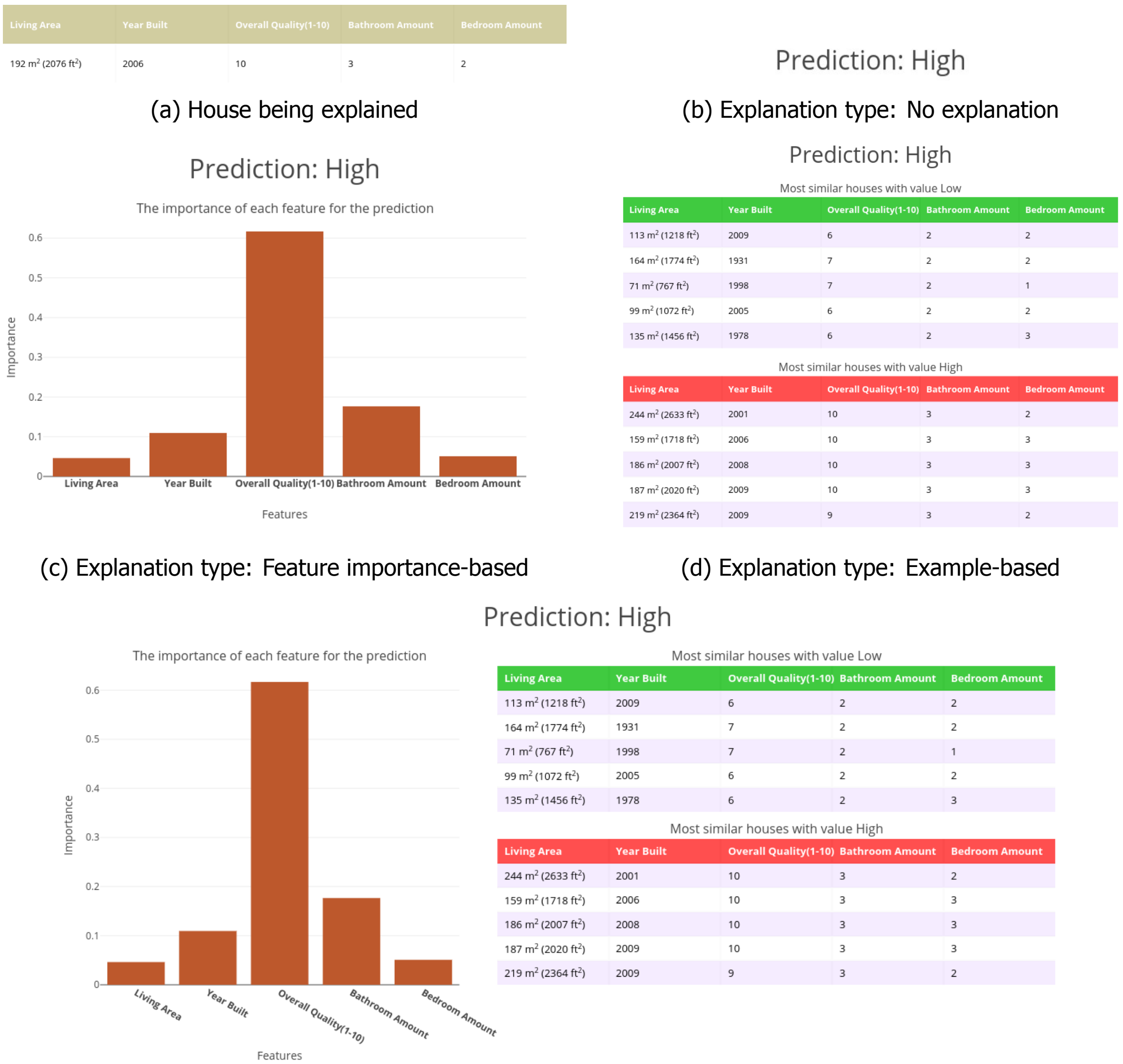
The dataset is split into two disjoint sets of training and testing set, which contain 70% and 30% of data, respectively. A SVM model with RBF kernel is built on the training set to predict sale value (low or high) of a house given its five features. This model has an AUC score of 98% on the testing set. The average local fidelity measure on the testing-set of LEAFAGE BIS strategy is equal to 98% with a standard deviation of 0.02%. In this user-study we use explanations extracted from LEAFAGE BIS strategy, because on average it performed better than LIME and LEAFAGE BDS on models with non-linear decision boundary.

5.2. Experimental Design

The objective of the experiment is to investigate the effect of example-based and feature importance-based explanations extracted from LEAFAGE on the perceived aid in decision making, acceptance and measured transparency.

5.2.1. Independent variable

We investigate 4 types of explanations namely *feature importance-based*, *example-based*, a combination of *example and feature importance-based* and finally providing no explanation as a baseline. An example of each explanation can be found in figure 5.1. Figure 5.1a shows an example of a house. Figures 5.1b, 5.1c, 5.1d, 5.1e show four types of explanations for the prediction of the house.



Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
192 m² (2076 ft²)	2006	10	3	2

(a) House being explained

Prediction: High

(b) Explanation type: No explanation

Prediction: High

The importance of each feature for the prediction

(c) Explanation type: Feature importance-based

Prediction: High

Most similar houses with value Low

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
113 m² (1218 ft²)	2009	6	2	2
164 m² (1774 ft²)	1931	7	2	2
71 m² (767 ft²)	1998	7	2	1
99 m² (1072 ft²)	2005	6	2	2
135 m² (1456 ft²)	1978	6	2	3

Most similar houses with value High

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
244 m² (2633 ft²)	2001	10	3	2
159 m² (1718 ft²)	2006	10	3	3
186 m² (2007 ft²)	2008	10	3	3
187 m² (2020 ft²)	2009	10	3	3
219 m² (2364 ft²)	2009	9	3	2

(d) Explanation type: Example-based

Prediction: High

The importance of each feature for the prediction

Most similar houses with value Low

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
113 m² (1218 ft²)	2009	6	2	2
164 m² (1774 ft²)	1931	7	2	2
71 m² (767 ft²)	1998	7	2	1
99 m² (1072 ft²)	2005	6	2	2
135 m² (1456 ft²)	1978	6	2	3

Most similar houses with value High

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
244 m² (2633 ft²)	2001	10	3	2
159 m² (1718 ft²)	2006	10	3	3
186 m² (2007 ft²)	2008	10	3	3
187 m² (2020 ft²)	2009	10	3	3
219 m² (2364 ft²)	2009	9	3	2

(e) Explanation type: Example and feature importance-based.

Figure 5.1: The first figure on the left shows the house being explained. The rest of the figures are the different types of explanations considered in the user-study.

5.2.2. Dependent variables

First, we look into how the explanations are perceived by the participants in terms of aid in decision-making. We hypothesize that providing explanations behind predictions, aid more in decision making than providing no explanation. Example-based reasoning is known to be used in problem-solving e.g.

diagnosing a patient [20]. But given LEAFAGE example-based explanation, it might not be easy to understand which features were important for the prediction. We further hypothesize that providing feature importance-based explanation in addition, will aid more in decision making. We propose four different variables that are important in an explanation such that the user can make an informed decision. The descriptions of these variables along with the null hypothesis are listed below.

1. Transparency: Whether the user understands the reasons behind a prediction
   $H_0 1$: *The median transparency score is the same for all explanation methods.*
2. Information sufficiency: Whether the user has enough information to make an informed decision
   $H_0 2$: *The median information sufficiency score is the same for all explanation methods.*
3. Competence: Whether the explanation corresponds to the user's own decision making logic
   $H_0 3$: *The median competence score is the same for all explanation methods.*
4. Confidence: Whether the explanation makes the user more confident about his/her decision
   $H_0 4$: *The median confidence score is the same for all explanation methods.*

Next, we look into whether the different explanation types have different affect on the acceptance of a prediction by the user. Example-based reasoning is heavily used in law for the goal of justifying arguments, positions and decisions [20]. We hypothesize that LEAFAGE example-based explanation is more persuasive than feature importance-based explanation. Further, we hypothesize that a *example and feature importance-based* explanation is more persuasive than each separate.
$H_0 5$: *There is no association between the different explanation types and acceptance of a prediction.*

The previous hypotheses are about how the user perceives an explanation. We would like to measure objectively in what degree an explanation type makes the underlying black-box ML model transparent. We establish whether there is an association between the different explanation types and the measured transparency with the following null hypothesis.
$H_0 6$: *There is no association between the different explanation types and measured transparency.*

An overview of all hypotheses can be found in figure 5.2. In the *aid decision making* part the participants are asked to rate an explanation in Likert scale [46] according to transparency, information sufficiency, competence and confidence. The chosen scale has five values namely strongly disagree, disagree, undecided, agree and strongly agree. In the *agreement* part the participants are asked whether they accept the prediction after seeing the explanation. At last, the transparency is objectively measured by testing the participant. The participant is shown another house in the neighbourhood of the house being explained. He/she has to indicate what the system would predict as the sale value of this new house given the explanation. The participant has the options of *low*, *high* or *I do not know*. If the participants are not sure about their answer they are recommended to choose *I do not know*.

5.2.3. Attention checks

To make sure that the participants are filling in the survey with concentration, we include two types of attention checks in the survey. First, before starting the survey the user is asked four basic questions about the different explanation methods. They should be able to answer those trivial questions if they have read the descriptions of the different explanation methods with concentration. Second, two extra example-based explanations are added that have an obvious and simple correct answer for the measured transparency part. The new house for which they have to indicate what the predicted sale value would be, is one of the houses from the example-based explanation. If the participant looks at the explanation with concentration he/she will be able to get the correct predicted sale value of the new house. Furthermore, in the beginning the participants are told that attention checks are included in the survey. They are also told that they will receive a bonus if they pass the attention checks.

5.2.4. Participants

The participants for this user-study were recruited from Amazon Mechanical Turk (MTurk). Participants received a monetary compensation for their time.
In total 114 participants completed the survey. In the attention checks the average score is equal to

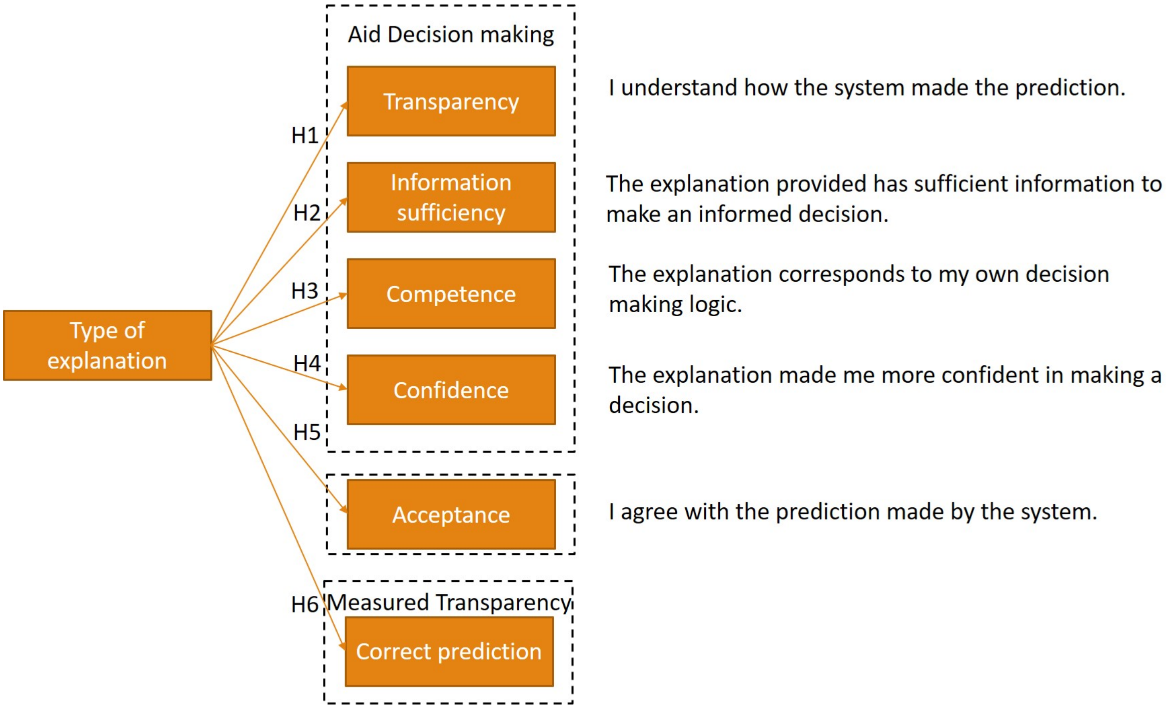

Aid Decision making
Transparency
Information sufficiency
Competence
Confidence
Acceptance
Measured Transparency
Correct prediction
Type of explanation
H1
H2
H3
H4
H5
H6
I understand how the system made the prediction.
The explanation provided has sufficient information to make an informed decision.
The explanation corresponds to my own decision making logic.
The explanation made me more confident in making a decision.
I agree with the prediction made by the system.

Figure 5.2: The dependent and independent variables.

3.55 with a standard deviation of 1.48. We assume that participants with a score smaller than 3 did not fill in the survey with sufficient concentration. We perform the analysis of 86 participants who scored at least 3 out of 6. The target group for this study is the general public. The demographics of the participants can be found in table 5.1. In terms of *Sex*, *Age* and *Highest level of education* the distributions seem well spread. Regarding *Region*, most of the participants are from the Americas. This could bias the results towards the preferences of the people of that region. Most of the participants are native or fluent English speakers. At last, 86% percentage of the participants have looked into buying a house.

5.2.5. Procedure

The different steps that the participants followed, are described in figure 5.3. Participants are first provided with a general introduction about the survey and asked to accept the terms of an informed consent. Next, they are provided with a demographic questionnaire. Further, the different types of explanations considered in this study are explained. Participants are then asked basic questions about the different explanation methods. The dependent variables described in figure 5.2 are measured next, on 42 explanations. Each participant saw 10 explanations per explanation type (40 explanations), plus two example-based explanation for attention check. The 42 houses being explained, were randomly chosen from the testing set. Moreover, all participants saw the same explanations in a randomized order. Before, the thank you and general remarks, the users are asked to provide their general perceptions for each type of explanation. The actual survey is available in the Appendix (chapter A).

5.3. Results

5.3.1. Descriptive statistics

Table 5.2 shows an overview of the means, including the standard deviation in brackets, of the score given by the participants on the dependent variables related to aid in decision making, for each explanation type. The mean score of all explanation types are higher than providing no explanation over all columns. Furthermore, example-based and the combination explanation have a higher mean score over all columns.

Sex		**Highest level of education**	
Male	45.3%	Less than high school	0.0%
Female	54.7%	High school	8.1%
Age		Some college (no degree)	15.1%
18 to 24	17.4%	Associate degree	15.1%
25 to 34	39.5%	Bachelor degree	48.8%
35 to 44	32.6%	Graduate degree	12.8%
45 to 54	17.4%	**Looked into buying a house**	
55 to 64	7.0%	Yes	86.0%
65 to 74	2.3%	No	14.0%
75+	0.0%	**Level of English**	
Region		Native/Fluent	89.5%
Africa	0.0%	Good	14.0%
Americas	80.2%	Satisfactory	0.0%
Asia	16.3%	Not very good	0.0%
Europe	3.5%	Bad	0.0%
Oceania	0.0%		

Table 5.1: Demographics information of the participants

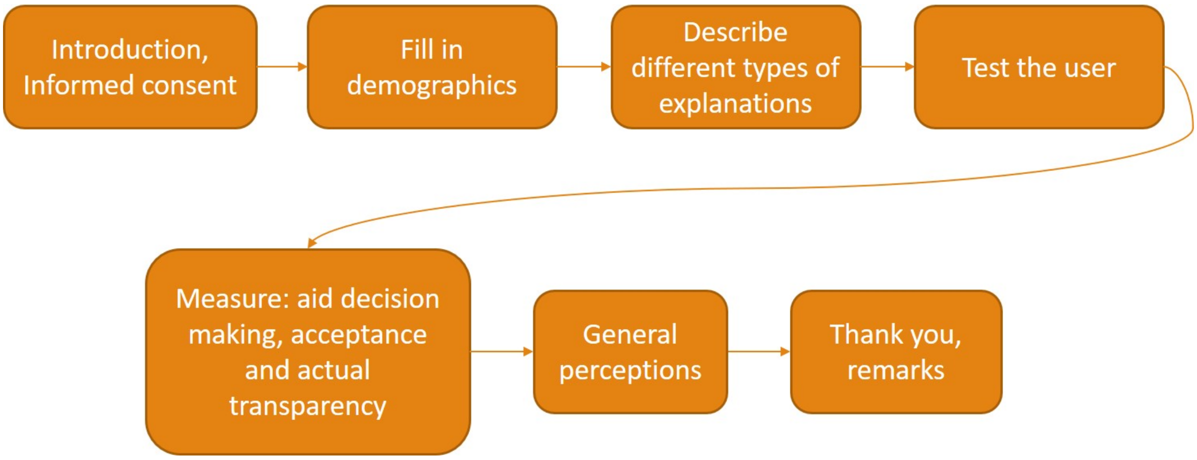

Introduction, Informed consent
Fill in demographics
Describe different types of explanations
Test the user
Measure: aid decision making, acceptance and actual transparency
General perceptions
Thank you, remarks

Figure 5.3: The procedure of the user-study.

	Transparency	Information Suff.	Competence	Confidence
No Explanation	3.66 (1.03)	3.43 (1.17)	3.70 (0.96)	3.52 (1.1)
Feature importance	3.92 (0.85)	3.76 (0.97)	3.78 (0.91)	3.68 (1.04)
Example-based	4.07 (0.76)	4.02 (0.84)	3.96 (0.86)	3.98 (0.86)
Example and Feature im.	4.13 (0.8)	4.10 (0.83)	3.93 (0.93)	3.98 (0.9)

Table 5.2: Means with standard deviation in brackets of the score given by the participants of the *aid decision making* part.

	I accept	I do not accept
No Explanation	748	112
Feature importance	736	124
Example-based	771	89
Example and Feature im.	751	109

Table 5.3: A cross table of the type of explanation and the acceptance by the user

	Correct Estimation	Wrong Estimation	I do not know
No Explanation	722	89	49
Feature importance	620	173	67
Example-based	759	88	13
Example and Feature im.	736	101	23

Table 5.4: A cross table of the type of explanation and the performance of the *measured transparency* part.

Next, table 5.3 shows a cross table of the type of explanation and the acceptance by the user. The predictions of the black-box model are accepted the most after participants were given example-based explanations. However, providing no explanation has more *I accept* than feature importance-based explanation.

At last, table 5.4 shows another cross table of the type of explanation and the performance of the *measured transparency* part. The *correct estimation* and *wrong estimation* columns lists the total amount of correct and wrong estimation of the sale value per explanation type, respectively. The participants were most uncertain about their answer and have the least amount of correct estimation after seeing a feature importance-based explanation. Furthermore, example-based explanations lead to the most amount of correct predictions and the least amount of *I do not know*.

5.3.2. Hypothesis testing

All of the hypotheses are tested with a significance level of $\alpha = 0.05$ with Bonferroni correction.

First, we test the four hypotheses related to aid in decision making. We verify whether there is a significant effect of the type of explanation on transparency, information sufficiency, competence and confidence. Four Kruskal-Wallis H-tests [47] are performed on each of those dependent variables. The results are visualized in table 5.5. We reject $H_0 1$, $H_0 2$, $H_0 3$ and $H_0 4$ with p-value < 0.001.
Furthermore, a Dunn's post-hoc test [48] is performed to look at significant differences ($\alpha = 0.0083$[1]) in performance between different explanation types. In terms of transparency, information sufficiency, competence and confidence both example-based and combination explanation perform significantly better than providing no explanation and feature importance-based explanation. No significant differences were found in performance between example-based and combination explanation regarding all four dependent variables. Feature-based explanation performs significantly better than providing no explanation in transparency, information sufficiency and confidence. But in terms of competence, feature-based does not perform significantly better than providing no explanation.

[1] Applying Bonferroni correction: 6 comparison per dependent variable leads to a significance level of $0.05/6 = 0.0083$

	Dependent Variable	H statistic	p-value
$H_0 1$	Transparency	124.22	< 0.001
$H_0 2$	Information Sufficiency	202.71	< 0.001
$H_0 3$	Competence	54.83	< 0.001
$H_0 4$	Confidence	125.24	< 0.001

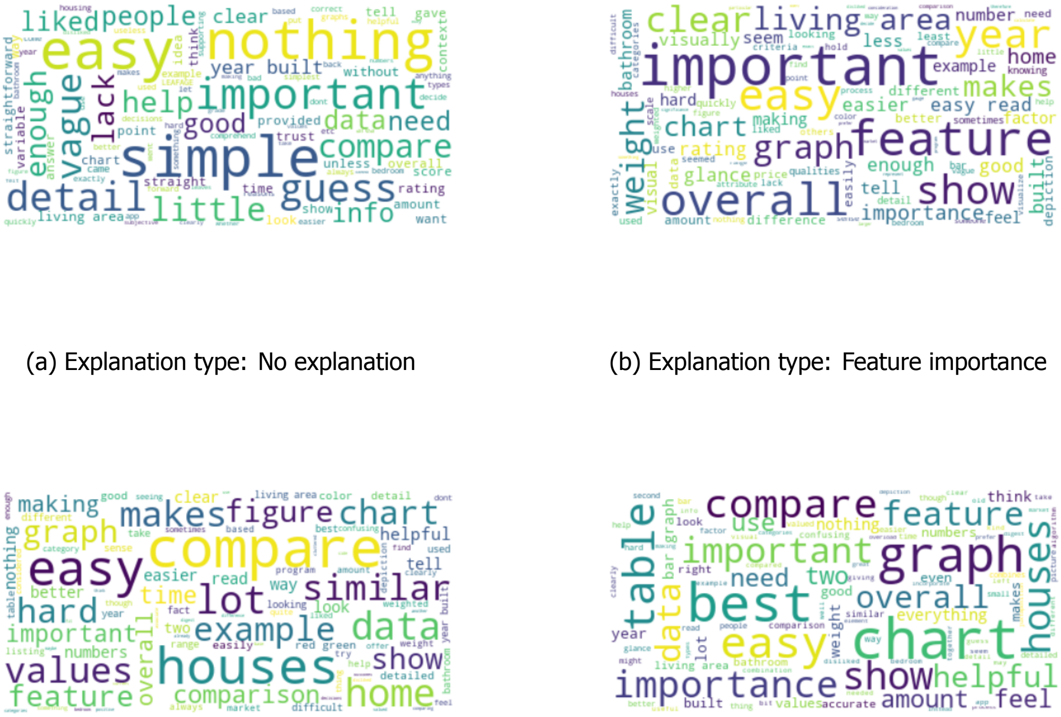
Table 5.5: Kruskal-Wallis H-tests are performed on the dependent variables related to aid in decision making.

Second, we verify whether there is a relation between the type of explanation and the acceptance of the prediction. A chi-square test of independence was performed to verify this relation. No significant relation was detected between the explanation type and acceptance, $\chi^2(3, N = 3440) = 6.67, p = 0.083$.

At last, a second chi-square test of independence was performed to verify an association between the type of explanation and measured transparency. For the analysis the amount of *Wrong Estimation* and *I do not know* are grouped together per explanation type. The result indicates that there is a significant association between those variables, $\chi^2(3, N = 3440) = 91.04, p < 0.001$. Post-hoc comparisons ($\alpha = 0.0083$) of the measured transparency by pairs of explanation types revealed that feature importance-based explanation lead to significantly less amount of correct estimation, compared to the rest of the explanation types. The amount of correct estimation was statistically similar between pairs of providing no explanation, example-based and combination explanation. It should be noted that between example-based explanation and providing no explanation the highest difference in correct estimation was measured with p-value=0.012.

5.3.3. General remarks about the explanation types

After the experiment the participants were asked to state their likes and dislikes for each of the explanation types. Figures 5.4a, 5.4b, 5.4c and 5.4d show word clouds of the free text given by the participants about each explanation type. These word clouds give a first glance of the impressions of the participants about the different types of explanation.



(a) Explanation type: No explanation

(b) Explanation type: Feature importance

(c) Explanation type: Example-based

(d) Example and feature-based explanation

Figure 5.4: Word clouds of the free text written by the participants about each explanation type.

The word cloud of providing no explanation (figure 5.4a) contains positive and less positive terms such as *simple*, *easy*, *vague* and *lack*. The answers show that the participants liked that the prediction was straightforward, easy to understand, simple, to the point and allowed them to make quick decisions. To quote a few: "Easy to digest and comprehend and conveys the key information that one wants to obtain", "I do not have to make any analysis", "No need to overthink" and "Easy to make a decision quickly". Furthermore, the participants disliked the lack of information supporting the prediction, the fact that the prediction provides no context and its vagueness. Here are a few quotes expressing those concerns: "Need complete trust in the system to find it helpful", "Could be seen as just a guess with no rational behind, doesn't seem as legitimate", "It felt like no information was used to give the prediction", "We should rely on the prediction made by the system but cannot verify the prediction" and "I don't want a simple rating without anything to back it up. Show me the numbers and let me decide what is important".

In the word cloud (figure 5.4b) of feature importance-based explanation, words related to the explanation type and some judgment stick out, such as *important*, *easy*, *feature* and *clear*. The participants found this explanation type straightforward, in showing how the sale value was determined and easy to read. They liked the visual aspect of the graph. The following quotes summarize their likes of the explanation type: "I like the easy to grasp and digest nature of this visual depiction of the rationale behind the rating used to evaluate a house." and "I like this type of explanation because it gives a graphical depiction of which features of the home is most important in the evaluation. This allows me to quickly look over different features and make a more informed choice." However the participants also had some concerns about this explanation type. They found this explanation type not detailed enough to perform well on the *measured transparency* part of the study. It was not clear how the importance really relates to the prediction of the house. Moreover, it was hard for them to understand the threshold of a feature which changes the prediction of a house. One participant wrote "If I had to list a dislike, it would be that there is no explanation WHY these features are of importance to the prediction model? How are they rated compared to one another? Do some features hold more weight overall? Because it seemed like it at times."

In the word cloud of example-based explanation (figure 5.4c), words about comparison and some contradicting judgments stand out such as *similar*, *compare*, *easy* and *hard*. The participants liked the fact that they could compare between similar houses with different sale values. They could see how the feature values differ between houses with different sale values. However, the participants both liked and disliked this explanation type because of the amount of information present in the tables. The large amount of raw data helped them to understand in detail why the prediction was made but lot of numbers were harder to digest and made the information look cluttered. Moreover, they found it hard to figure out which features were important. Two participants summarized the pros and cons as follows: "I like this explanation because it provides references as to what would be considered a high or low valued home. With this layout I can easily compare the target home to similar homes on the market and decipher what features are associated with the high or low value homes. The only downside this explanation is that he does not provide a detailed explanation as to the importance of each feature in making the decision." and "I think the best thing about this type of explanation is that it is very quick and easy to compare multiple models at the same time. It is much easier to see how the features compare to one another here. And by looking at that, I get a better idea of why the prediction was made. I suppose a con could be too much information at once."

The word cloud (figure 5.4d) of example and feature importance-based explanation shows words related to the previous two explanation types and one word in particular namely *best* stands out. The participants liked the combination of example-based and feature importance-based explanation but also stated that the amount of information can be overwhelming. Some participants said that they only looked at one chart and ignored the other. The overall remarks of this explanation type is nicely summarized by one participant as follows: "I like this explanation type because it incorporates both an easy to digest visual depiction i.e. the bar graph and the raw data used to build this algorithm calculator. This juxtaposition of key elements of the algorithm facilitates the ability of users to obtain a more detailed and informed idea regarding the backbone of the algorithm. I guess one potential downside is that low information users may be turned off and/or intimidated by the juxtaposition of a graph and data chart."

At last, the participants were asked to give general remarks about the survey. The participants stated that even though the survey was long they could stay engaged because of the *measured transparency* part. One participant wrote: “The survey was lengthy, but engaging and fun to participate in. I strongly believe that information like this will be a great benefit to home buyers.”

6

Conclusion

In this thesis we developed a new method called LEAFAGE that provides contrastive example and feature-based explanations for the predictions made by a black-box ML model. Two different sampling strategies to build the local interpretable model have been defined in the workings of LEAFAGE, namely: Boundary Independent Sampling (BIS) and Boundary Dependent Sampling (BDS). Furthermore, for the evaluation of LEAFAGE two evaluation methods were used. The first method evaluated whether LEAFAGE explanations reflect the true reasoning of the underlying black-box ML model (*local fidelity*). Second, we looked into the usefulness of the explanations from the user's point of view, through conducting a user-study.

We proposed a new method to evaluate the local fidelity of a local interpretable model that mimics the black-box classifier, in the neighbourhood of the instance being explained. This evaluation method was used to evaluate the BIS and BDS methods, along with the current state-of-art method LIME [1]. The evaluation was done for 4 datasets with different characteristics and 6 different ML models. No clear winner among the three strategies was found. Overall LIME performed better than the other strategies, on models with linear decision boundaries. On the other hand, BIS and BDS performed overall better than LIME on non-linear models. Between BIS and BDS, BIS performed better on non-linear models.

By performing a user-study, we evaluated the usefulness of example-based and feature importance-based explanations extracted from LEAFAGE, in terms of the perceived aid in decision making, acceptance and measured transparency. The context of the user-study was to help the participants in estimating the value of a house, by providing a prediction made by a ML model and an explanation. Example-based explanation performed significantly better than providing no explanation and feature importance-based explanation, regarding perceived transparency, information sufficiency, competence and confidence. This was expected because example-based reasoning is regarded as a natural way of solving problems in our daily lives [22, 23]. Moreover, we had expected that adding information about the importance of features would increase the perceived aid in decision making, but no significant difference was found. This could be because the amount of information became overwhelming and seemed cluttered as the comments of the participants suggested. Feature importance-based explanation performed significantly better than providing no explanation in terms of transparency, information sufficiency and confidence. However no significant difference with providing no explanation was detected regarding competence. This result suggests that feature importance-based explanation does not align more with the decision making of the participants than providing no explanation.

We expected a significant difference between example-based explanation and providing no explanation in relation to acceptance, because example-based reasoning is generally used to justify arguments and convince judges in court [20]. However, no significant relation was detected between the explanation types and acceptance. This could be because even without any explanations, the participants already had a judgment of the house because most of them indicated that they have been looking into buying a house in the past. The prediction could have validated their judgment which led to acceptance of the prediction, even without seeing an explanation. We suspect that a significant difference can be measured in a use-case in which it is hard to make a decision after only looking at an instance.

At last, we expected that example-based explanations would lead to higher measured transparency than providing no explanation but no significant association was found. Moreover, feature importance-based explanation led to significantly less amount of correct estimation, compared to the rest of the explanation types. These results suggest that there is a discrepancy between the perceived transparency and the measured transparency, because the participants preferred both feature-importance and example-based explanation over providing no explanation. This could be because the task was too easy and people were able to guess the sale value of the house even without seeing an explanation. The fact that the participants performed worse after seeing a feature importance-based explanation than providing no explanation, suggests that feature importance-based explanations confuse the users in how they can generalize the provided classification logic in the explanation to other instances. In the comments some participants noted that it was not clear how the importance of a feature really relates to the prediction of the house.

Furthermore, we list some possible interesting directions for future research. We currently only use the training instances to build a local linear model. If an instance to be explained is an outlier, the corresponding LEAFAGE explanation about its prediction might no be locally faithful to the underlying model. For future work, we can study the use of generated artificial samples which enrich the training instances such that even for outlier instances, faithful explanations can be extracted.

LEAFAGE is able to generate example-based explanations only of tabular datasets. It is an interesting direction of research to expand it to text and image data. Especially, providing images as examples could be a lot easier to understand than cluttered tabular data.

In conclusion, LEAFAGE explanation performed overall better than the current state-of-the-art method LIME on non-linear models, in terms of local fidelity. The empirical evaluation showed that overall the participants perceived receiving explanations behind a prediction as more helpful than providing no explanation for the goal of decision making. However when the participants were tested about their gained knowledge after seeing an explanation, no significant advantage was found compared to providing no explanation. We suspect that this is due to the simplicity of the test, as future work a more comprehensive test could be used to measure the actual transparency. Furthermore, example-based explanation significantly performed better than feature-importance based explanation in terms of aid in decision making. Showing both example-based and feature-importance-based explanation did not increase the perceived aid in decision making, significantly. This could be due to the overload of information as the participants described.

Bibliography

[1] 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.

[2] Z. C. Lipton, *The mythos of model interpretability,* arXiv preprint arXiv:1606.03490 (2016).

[3] S. Kaufman, S. Rosset, C. Perlich, and O. Stitelman, *Leakage in data mining: Formulation, detection, and avoidance,* ACM Transactions on Knowledge Discovery from Data (TKDD) **6**, 15 (2012).

[4] R. Kohavi, C. E. Brodley, B. Frasca, L. Mason, and Z. Zheng, *Kdd-cup 2000 organizers' report: Peeling the onion,* ACM SIGKDD Explorations Newsletter **2**, 86 (2000).

[5] J. Quionero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence, *Dataset shift in machine learning* (The MIT Press, 2009).

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

[7] K. Patel, J. Fogarty, J. A. Landay, and B. Harrison, *Investigating statistical machine learning as a tool for software development,* in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (ACM, 2008) pp. 667–676.

[8] S. I. Ross Casey, *Ibm pitched its watson supercomputer as a revolution in cancer care. it's nowhere close,* Website.

[9] A. Chouldechova, *Fair prediction with disparate impact: A study of bias in recidivism prediction instruments,* Big data **5**, 153 (2017).

[10] S. Barocas and A. D. Selbst, *Big data's disparate impact,* Cal. L. Rev. **104**, 671 (2016).

[11] B. Goodman and S. Flaxman, *European union regulations on algorithmic decision-making and a" right to explanation",* arXiv preprint arXiv:1606.08813 (2016).

[12] M. Leese, *The new profiling: Algorithms, black boxes, and the failure of anti-discriminatory safeguards in the european union,* Security Dialogue **45**, 494 (2014).

[13] M. Kosinski, D. Stillwell, and T. Graepel, *Private traits and attributes are predictable from digital records of human behavior,* Proceedings of the National Academy of Sciences **110**, 5802 (2013).

[14] P. Lipton, *Contrastive explanation,* Royal Institute of Philosophy Supplements **27**, 247 (1990).

[15] T. Miller, *Explanation in artificial intelligence: Insights from the social sciences,* arXiv preprint arXiv:1706.07269 (2017).

[16] S. Begum, M. U. Ahmed, P. Funk, N. Xiong, and M. Folke, *Case-based reasoning systems in the health sciences: a survey of recent trends and developments,* IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) **41**, 421 (2011).

[17] B. Kim, C. Rudin, and J. A. Shah, *The bayesian case model: A generative approach for case-based reasoning and prototype classification,* in *Advances in Neural Information Processing Systems* (2014) pp. 1952–1960.

[18] H. A. Simon and A. Newell, *Human problem solving: The state of the theory in 1970.* American Psychologist **26**, 145 (1971).

[19] T. Miller, P. Howe, and L. Sonenberg, *Explainable ai: Beware of inmates running the asylum,* in *IJCAI-17 Workshop on Explainable AI (XAI)* (2017) p. 36.

[20] J. L. Kolodner, *An introduction to case-based reasoning,* Artificial intelligence review **6**, 3 (1992).

[21] M. M. Richter and R. O. Weber, *Case-based reasoning* (Springer, 2016).

[22] I. Bichindaritz and C. Marling, *Case-based reasoning in the health sciences: What's next?* Artificial intelligence in medicine **36**, 127 (2006).

[23] A. Aamodt and E. Plaza, *Case-based reasoning: Foundational issues, methodological variations, and system approaches,* AI communications **7**, 39 (1994).

[24] D. J. Hilton and B. R. Slugoski, *Knowledge-based causal attribution: The abnormal conditions focus model.* Psychological review **93**, 75 (1986).

[25] D. J. Hilton and L. M. JOHN, *The course of events: counterfactuals, causal sequences, and explanation,* in *The psychology of counterfactual thinking* (Routledge, 2007) pp. 56–72.

[26] H. Chockler and J. Y. Halpern, *Responsibility and blame: A structural-model approach,* Journal of Artificial Intelligence Research **22**, 93 (2004).

[27] J. Leddo, R. P. Abelson, and P. H. Gross, *Conjunctive explanations: When two reasons are better than one.* Journal of Personality and Social Psychology **47**, 933 (1984).

[28] N. Tintarev and J. Masthoff, *Designing and evaluating explanations for recommender systems,* in *Recommender systems handbook* (Springer, 2011) pp. 479–510.

[29] P. Pu, L. Chen, and R. Hu, *A user-centric evaluation framework for recommender systems,* in *Proceedings of the fifth ACM conference on Recommender systems* (ACM, 2011) pp. 157–164.

[30] F. Gedikli, D. Jannach, and M. Ge, *How should i explain? a comparison of different explanation types for recommender systems,* International Journal of Human-Computer Studies **72**, 367 (2014).

[31] H. Cramer, V. Evers, S. Ramlal, M. Van Someren, L. Rutledge, N. Stash, L. Aroyo, and B. Wielinga, *The effects of transparency on trust in and acceptance of a content-based art recommender,* User Modeling and User-Adapted Interaction **18**, 455 (2008).

[32] P. Pu and L. Chen, *Trust-inspiring explanation interfaces for recommender systems,* Knowledge-Based Systems **20**, 542 (2007).

[33] P. Gill, *Introduction to Machine Learning Interpretability* (O'Reilly Media, Incorporated, 2018).

[34] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, *Show, attend and tell: Neural image caption generation with visual attention,* in *International conference on machine learning* (2015) pp. 2048–2057.

[35] T. Laugel, X. Renard, M.-J. Lesot, C. Marsala, and M. Detyniecki, *Defining locality for surrogates in post-hoc interpretablity,* arXiv preprint arXiv:1806.07498 (2018).

[36] R. Guidotti, A. Monreale, F. Turini, D. Pedreschi, and F. Giannotti, *A survey of methods for explaining black box models,* arXiv preprint arXiv:1802.01933 (2018).

[37] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)* (Springer-Verlag, Berlin, Heidelberg, 2006).

[38] C. Molnar, *Interpretable Machine Learning* (https://christophm.github.io/interpretable-ml-book/, 2018) `https://christophm.github.io/interpretable-ml-book/`.

[39] M. T. Ribeiro, *Github code-base of lime,* Website.

[40] T. Laugel, M.-J. Lesot, C. Marsala, X. Renard, and M. Detyniecki, *Comparison-based inverse classification for interpretability in machine learning,* in *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems* (Springer, 2018) pp. 100–111.

[41] R. Fisher, *Iris dataset,* Website.

[42] M. Forina, *Wine dataset,* Website.

[43] W. H. Wolberg, W. Nick, and O. L. Mangasarian, *Breast cancer dataset,* Website.

[44] V. Lohweg, *Banknote authentication dataset,* Website.

[45] D. De Cock, *Ames, iowa: Alternative to the boston housing data as an end of semester regression project,* Journal of Statistics Education **19** (2011).

[46] G. Norman, *Likert scales, levels of measurement and the "laws" of statistics,* Advances in health sciences education **15**, 625 (2010).

[47] W. H. Kruskal and W. A. Wallis, *Use of ranks in one-criterion variance analysis,* Journal of the American statistical Association **47**, 583 (1952).

[48] O. J. Dunn, *Multiple comparisons using rank sums,* Technometrics **6**, 241 (1964).

A

Appendix

A.1. Survey

Introduction

Thank you for your interest in this online survey. The survey will take approximately 40 minutes.

Buying a house
Imagine you are looking to buy a property. You are searching online and you find a couple of houses you like. Because it is a big investment, you are interested in the real value of the house. You find a smart application called LEAFAGE that estimates the value of a house automatically given its different features. For simplicity, let’s say it predicts whether a property has low or high value. LEAFAGE provides four types of explanation, indicating why the house was predicted with a certain value. In this study we want to understand which type of these explanations are useful to the user in decision making.

Experiment
You will be shown 42 houses, along with its predicted values and explanations made by the smart application. For each house you have to rate the explanation and indicate whether you agree with the prediction. Further, to assess whether the explanation is clear, you will be shown a similar house and you have to indicate what the smart application would predict as the sale value. At last, after your input on the 42 houses, you will receive questions about your general perceptions of the different types of explanations.

Informed Consent

Before taking part in this study, please read the consent form below and and continue to the next page if you understand the statements and freely consent to participate in the study.

Your responses will be collected anonymously and are kept confidential. There are no known or anticipated risks associated with this study. If participants have further questions about this study or their rights, or if they wish to lodge a complaint or concern, they may contact the researchers at ajaya.adhikari[at]tno.nl

By continuing to the next page you indicate that you are at least 18 years old, have read and understood this consent, and voluntarily participate in this study.

Payment

At the end of the survey you will receive a code, that you can enter in Amazon Mechanical Turk. In that way we can verify that you have participated in the survey. You will be awarded $4 through Amazon Mechanical Turk. You are free to withdraw and stop at any moment of the experiment. If you decide to do so, you will not be paid for the survey.

Before approving your payment to Amazon Mechanical Turk, your responses will be first checked to make sure that you have filled in the survey with concentration. There are a few attention checks through out the survey. If you do not pass any of those checks you will not be paid. And if you pass all of these tests you will gain a bonus of $2.

Demographics

What is your sex?

○ Male

○ Female

What is your age?

○ 18 to 24

○ 25 to 34

○ 35 to 44

○ 45 to 54

○ 55 to 64

○ 65 to 74

○ 75 or older

In what region do you currently reside?

○ Africa

○ Americas

○ Asia

○ Europe

○ Oceania

What is the highest level of school you have completed or the highest degree you have received?

○ Less than high school degree

○ High school degree or equivalent (e.g., GED)

○ Some college but no degree

○ Associate degree

○ Bachelor degree

○ Graduate degree

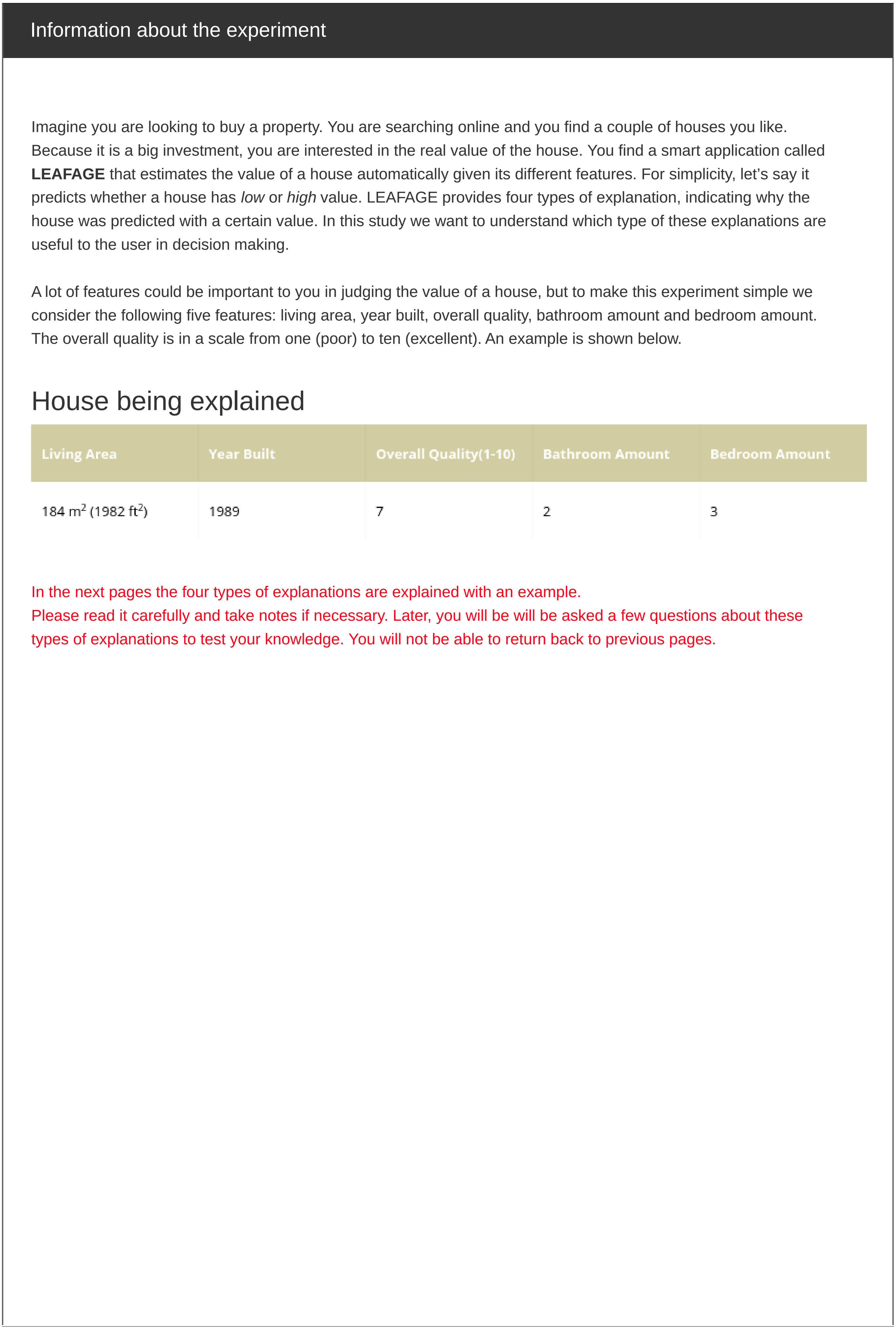
Have you ever bought or looked into buying a house before?

○ Yes

○ No

What is your level of English?

Bad	Not very good	Satisfactory	Good	Native/Fluent
○	○	○	○	○



Information about the experiment

Imagine you are looking to buy a property. You are searching online and you find a couple of houses you like. Because it is a big investment, you are interested in the real value of the house. You find a smart application called **LEAFAGE** that estimates the value of a house automatically given its different features. For simplicity, let's say it predicts whether a house has *low* or *high* value. LEAFAGE provides four types of explanation, indicating why the house was predicted with a certain value. In this study we want to understand which type of these explanations are useful to the user in decision making.

A lot of features could be important to you in judging the value of a house, but to make this experiment simple we consider the following five features: living area, year built, overall quality, bathroom amount and bedroom amount. The overall quality is in a scale from one (poor) to ten (excellent). An example is shown below.

House being explained

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
184 m^2 (1982 ft^2)	1989	7	2	3

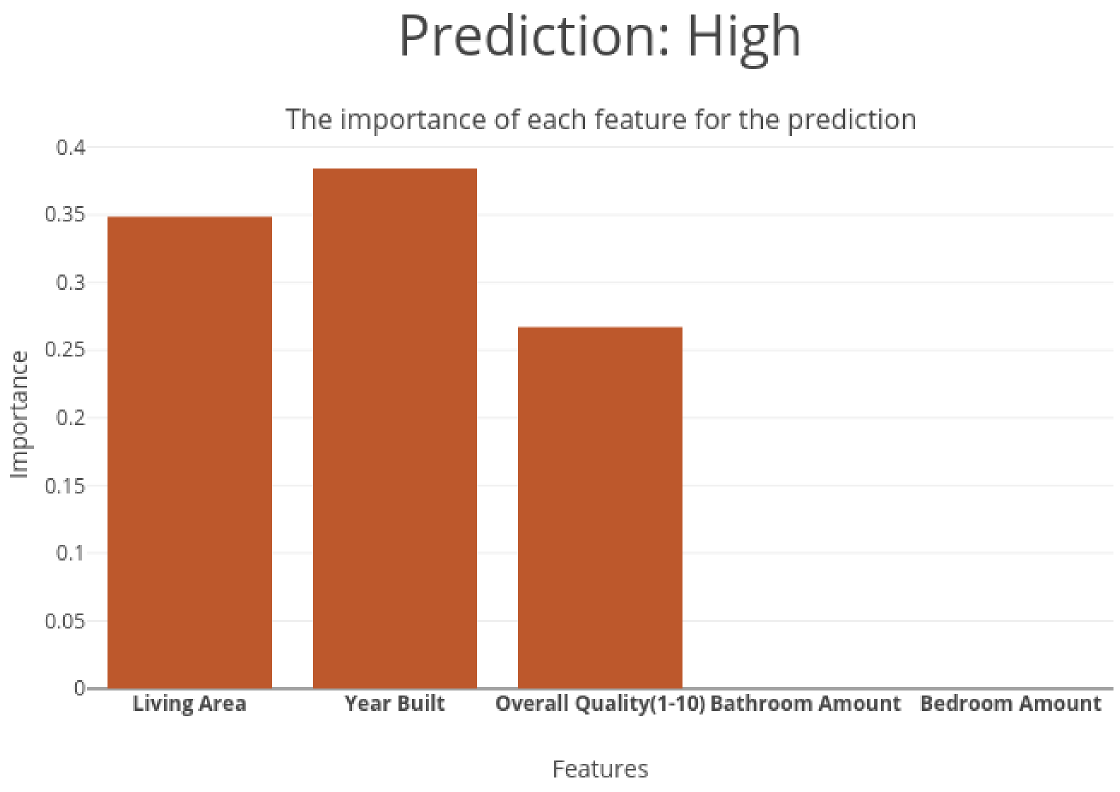
In the next pages the four types of explanations are explained with an example.
Please read it carefully and take notes if necessary. Later, you will be will be asked a few questions about these types of explanations to test your knowledge. You will not be able to return back to previous pages.

Explanation type

House being explained

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
184 m^2 (1982 ft^2)	1989	7	2	3

Output of LEAFAGE



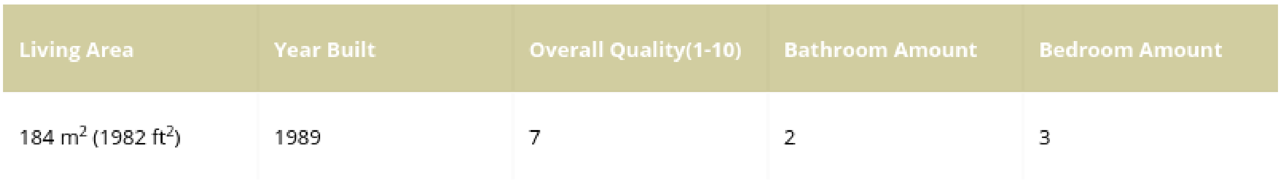
Prediction: High

The importance of each feature for the prediction

This explanation type first shows the predicted value of the house. Furthermore, it shows which of the features of the house were most important to make the prediction. The length of each bar shows the importance of the feature. In this case, the features *Bathroom Amount=2* and *Bedroom Amount=3* are not important. Feature *Year Built=1989* is the most important followed by *Living Area=184m*^{2} and *Overall Quality=7.*

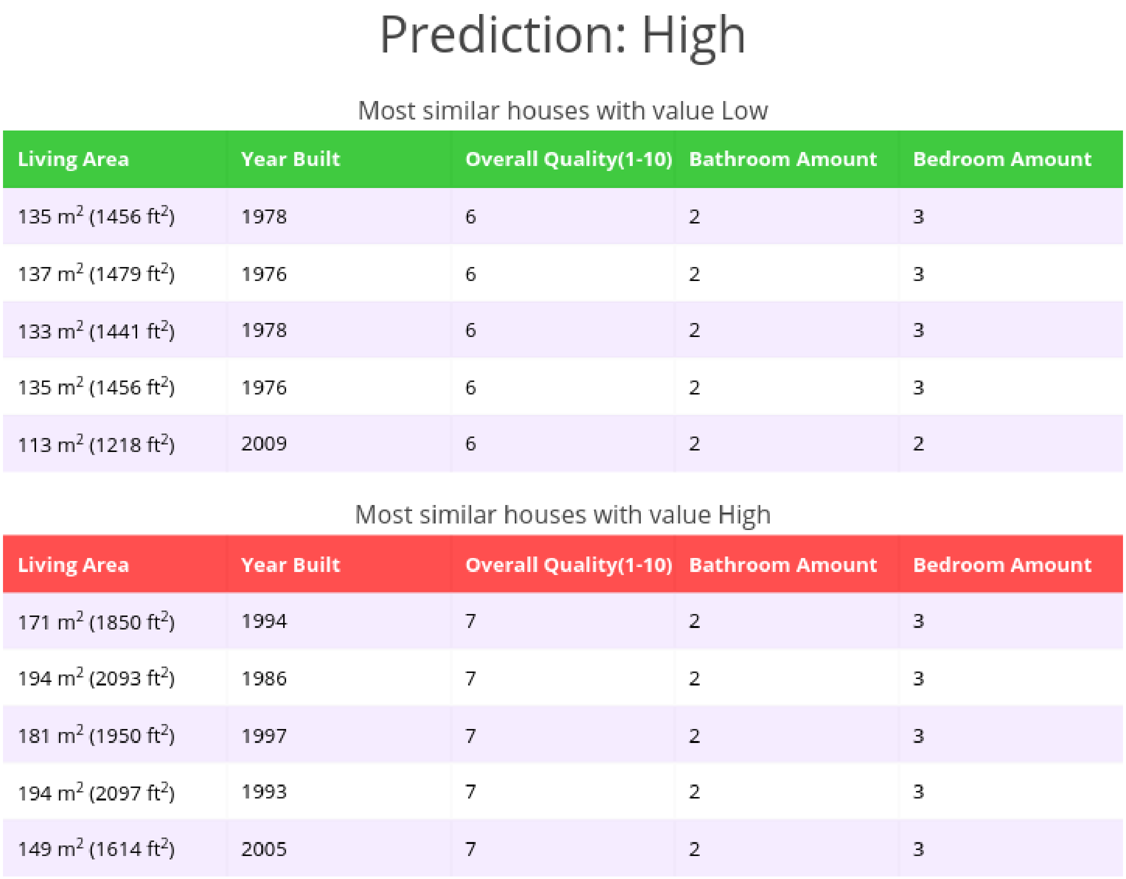
Explanation type

House being explained



Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
184 m^2 (1982 ft^2)	1989	7	2	3

Output of LEAFAGE



Prediction: High

Most similar houses with value Low

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
135 m^2 (1456 ft^2)	1978	6	2	3
137 m^2 (1479 ft^2)	1976	6	2	3
133 m^2 (1441 ft^2)	1978	6	2	3
135 m^2 (1456 ft^2)	1976	6	2	3
113 m^2 (1218 ft^2)	2009	6	2	2

Most similar houses with value High

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
171 m^2 (1850 ft^2)	1994	7	2	3
194 m^2 (2093 ft^2)	1986	7	2	3
181 m^2 (1950 ft^2)	1997	7	2	3
194 m^2 (2097 ft^2)	1993	7	2	3
149 m^2 (1614 ft^2)	2005	7	2	3

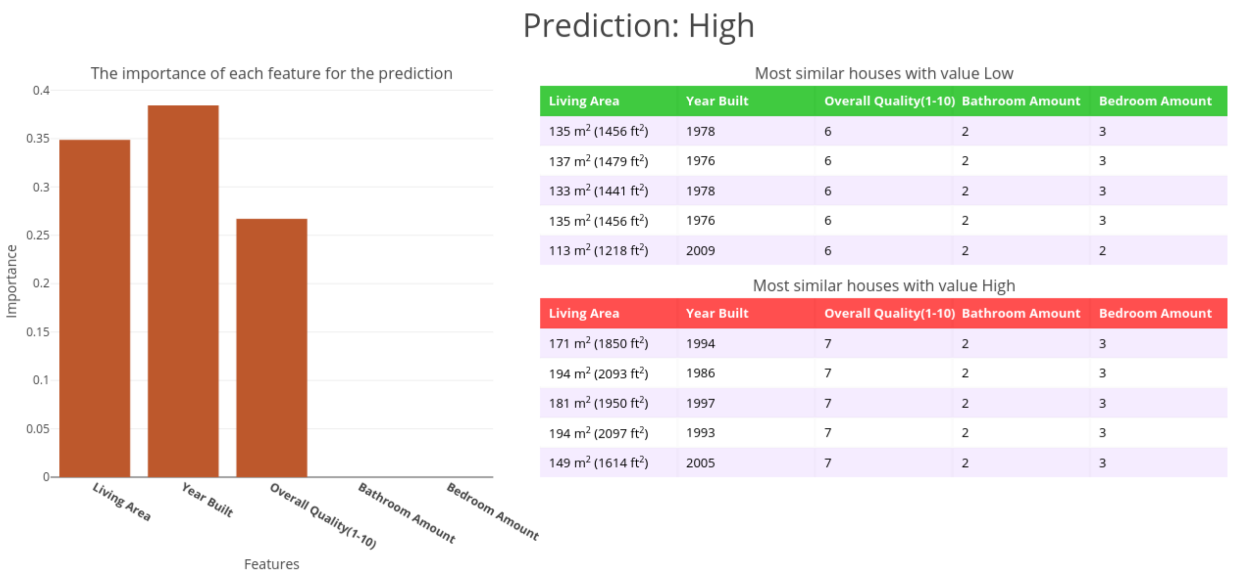
This explanation type first shows the predicted value of the house. Furthermore, it shows houses that are similar to the house being explained. The green table shows 5 similar houses that have low value and the red table shows 5 similar houses with high value. These tables make clear how big the differences for each feature is, between similar houses with value low and high. For example in this explanation, the differences in *Living Area* are big between low and high value houses while there are no differences in *Bathroom Amount* and *Bedroom Amount*.

Explanation type

House being explained

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
184 m^2 (1982 ft^2)	1989	7	2	3

Output of LEAFAGE



Prediction: High

Most similar houses with value Low

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
135 m^2 (1456 ft^2)	1978	6	2	3
137 m^2 (1479 ft^2)	1976	6	2	3
133 m^2 (1441 ft^2)	1978	6	2	3
135 m^2 (1456 ft^2)	1976	6	2	3
113 m^2 (1218 ft^2)	2009	6	2	2

Most similar houses with value High

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
171 m^2 (1850 ft^2)	1994	7	2	3
194 m^2 (2093 ft^2)	1986	7	2	3
181 m^2 (1950 ft^2)	1997	7	2	3
194 m^2 (2097 ft^2)	1993	7	2	3
149 m^2 (1614 ft^2)	2005	7	2	3

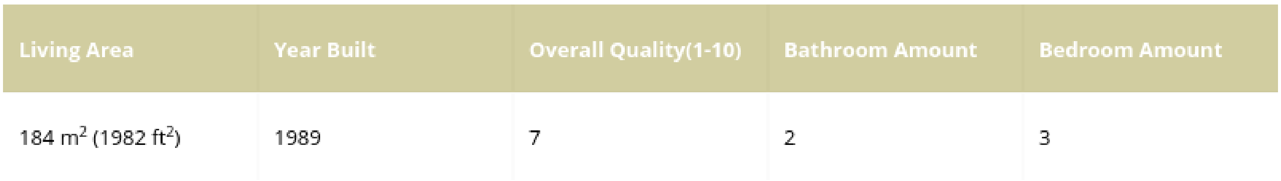
This explanation type first shows the predicted value of the house. Furthermore, it combines the two types of explanation.

The left graph shows which of the features of the house were the most important to make the prediction. The length of each bar shows the importance of the feature. In this case, the features *Bathroom Amount=2* and *Bedroom Amount=3* are not important. Feature *Year Built=1989* is the most important followed by *Living Area=184m^2* and *Overall Quality=7*.

The two tables in the right show houses that are similar to the house being explained. The green table shows 5 similar houses that have low value and the red table shows 5 similar houses with high value. These tables make clear how big the differences for each feature are, between similar houses with value low and high. For example in this explanation, the differences in *Living Area* are big between low and high value houses while there are no differences in *Bathroom Amount* and *Bedroom Amount.*

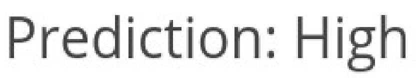
Explanation Type

House being explained



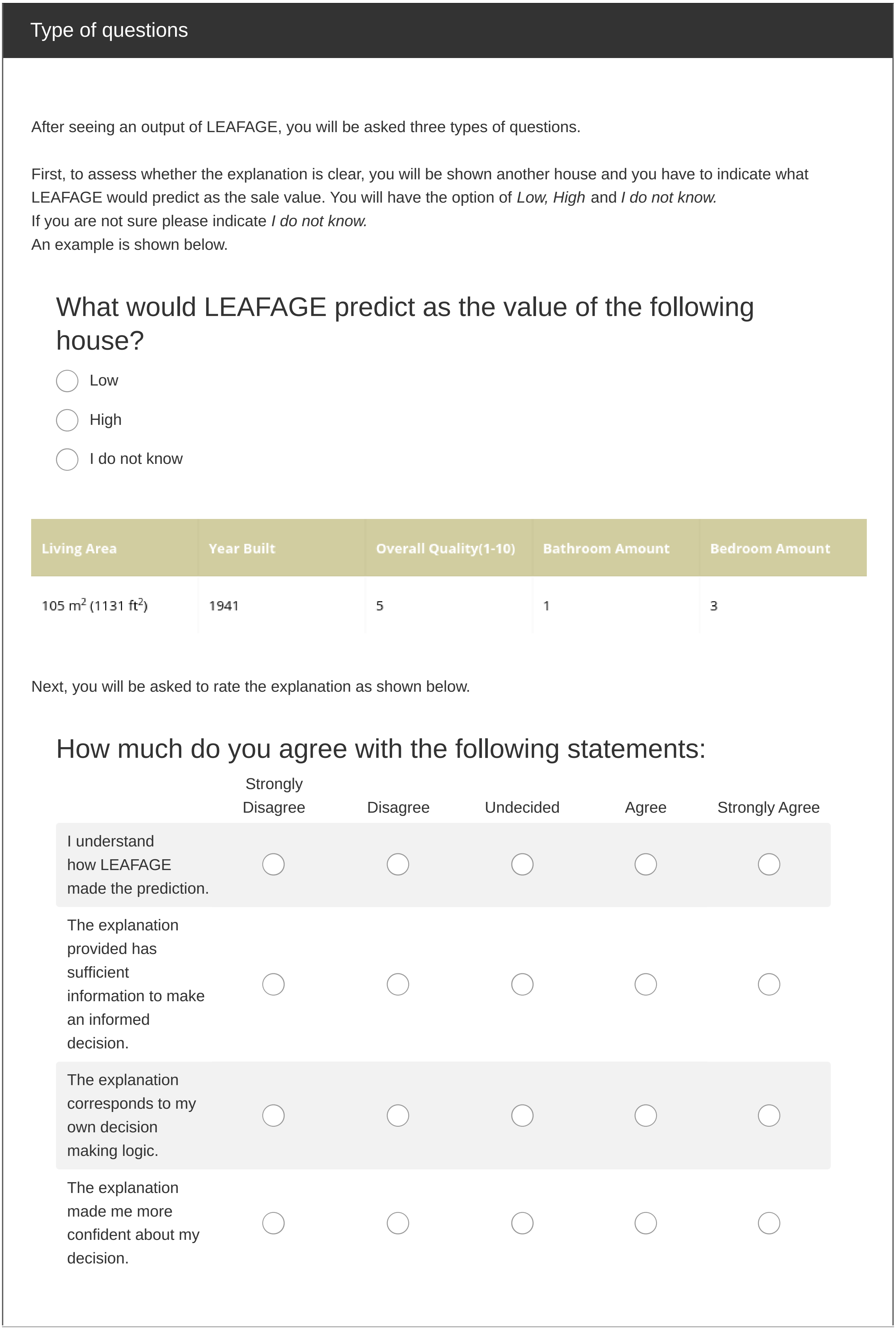
Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
184 m^2 (1982 ft^2)	1989	7	2	3

Output of LEAFAGE



Prediction: High

This explanation type only shows the predicted value of the house.



Type of questions

After seeing an output of LEAFAGE, you will be asked three types of questions.

First, to assess whether the explanation is clear, you will be shown another house and you have to indicate what LEAFAGE would predict as the sale value. You will have the option of *Low, High* and *I do not know.*
If you are not sure please indicate *I do not know.*
An example is shown below.

What would LEAFAGE predict as the value of the following house?

○ Low

○ High

○ I do not know

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
105 m^2 (1131 ft^2)	1941	5	1	3

Next, you will be asked to rate the explanation as shown below.

How much do you agree with the following statements:

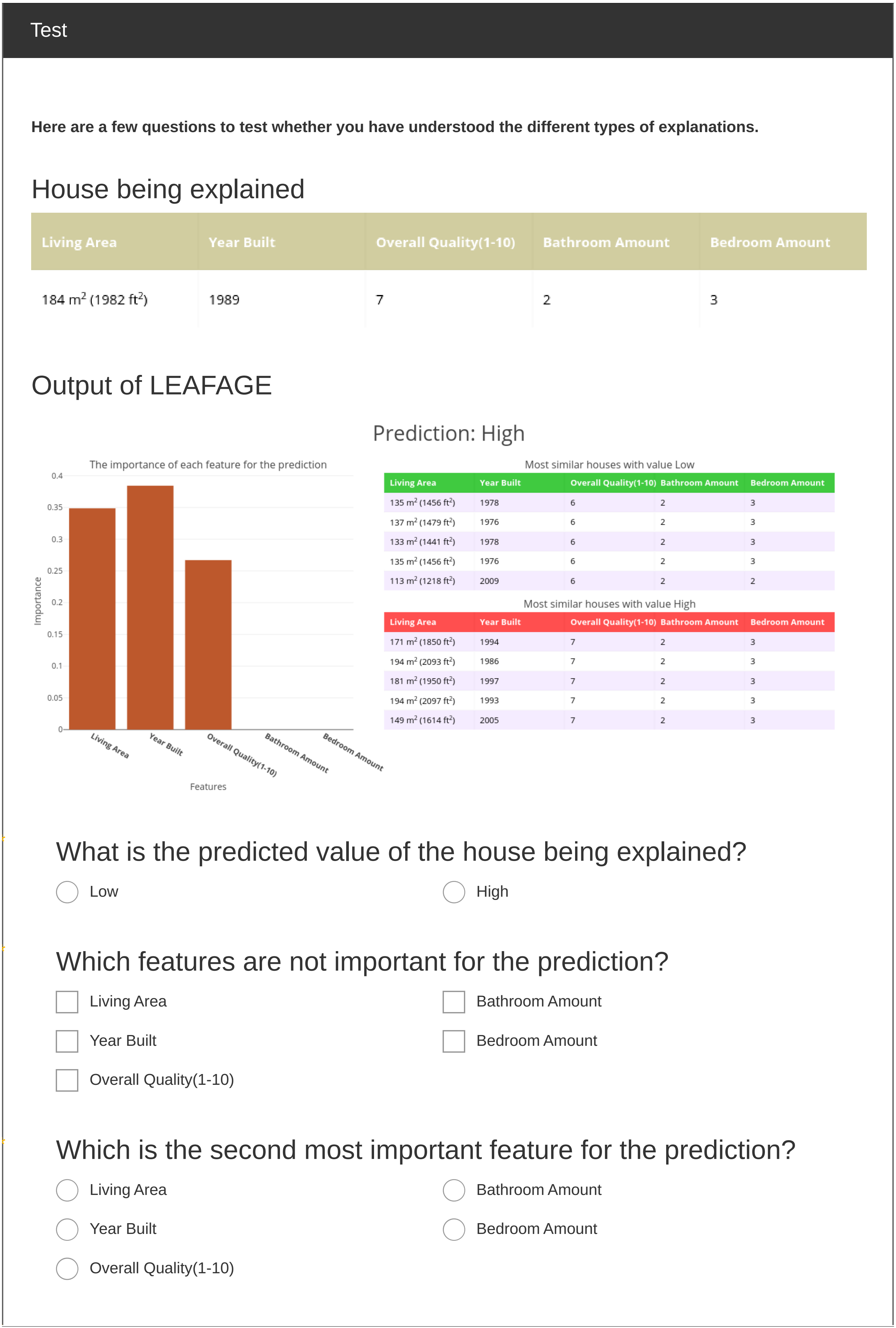
	Strongly Disagree	Disagree	Undecided	Agree	Strongly Agree
I understand how LEAFAGE made the prediction.	○	○	○	○	○
The explanation provided has sufficient information to make an informed decision.	○	○	○	○	○
The explanation corresponds to my own decision making logic.	○	○	○	○	○
The explanation made me more confident about my decision.	○	○	○	○	○

At last, you will be asked to indicate whether you accept the prediction after seeing the output of LEAFAGE.
An example is shown below.

Do you accept the prediction made by LEAFAGE?

○ Yes

○ No



Test

Here are a few questions to test whether you have understood the different types of explanations.

House being explained

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
184 m^2 (1982 ft^2)	1989	7	2	3

Output of LEAFAGE

Prediction: High

Most similar houses with value Low

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
135 m^2 (1456 ft^2)	1978	6	2	3
137 m^2 (1479 ft^2)	1976	6	2	3
133 m^2 (1441 ft^2)	1978	6	2	3
135 m^2 (1456 ft^2)	1976	6	2	3
113 m^2 (1218 ft^2)	2009	6	2	2

Most similar houses with value High

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
171 m^2 (1850 ft^2)	1994	7	2	3
194 m^2 (2093 ft^2)	1986	7	2	3
181 m^2 (1950 ft^2)	1997	7	2	3
194 m^2 (2097 ft^2)	1993	7	2	3
149 m^2 (1614 ft^2)	2005	7	2	3

What is the predicted value of the house being explained?

○ Low ○ High

Which features are not important for the prediction?

□ Living Area □ Bathroom Amount

□ Year Built □ Bedroom Amount

□ Overall Quality(1-10)

Which is the second most important feature for the prediction?

○ Living Area ○ Bathroom Amount

○ Year Built ○ Bedroom Amount

○ Overall Quality(1-10)

Between the low and high houses, which features have different values?

□ Living Area

□ Year Built

□ Overall Quality(1-10)

□ Bathroom Amount

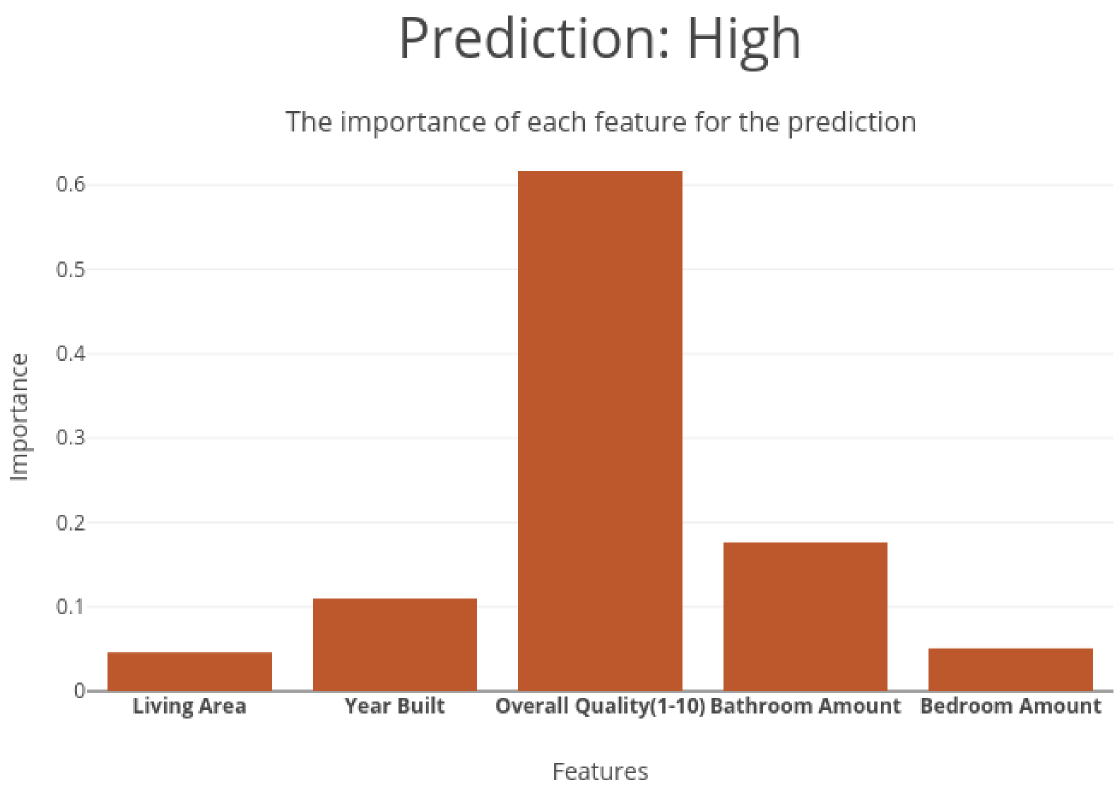
□ Bedroom Amount

Start experiment

Click next to start with the experiment. Please fill in the questions with concentration. Attention checks are included in the survey.

Final Questions

Now, you can provide your opinion on the different types of explanation in free text. Please state what you like and dislike about the different types for the goal of decision making.



What are your likes and dislikes in the above type of explanation?

Prediction: High

Most similar houses with value Low

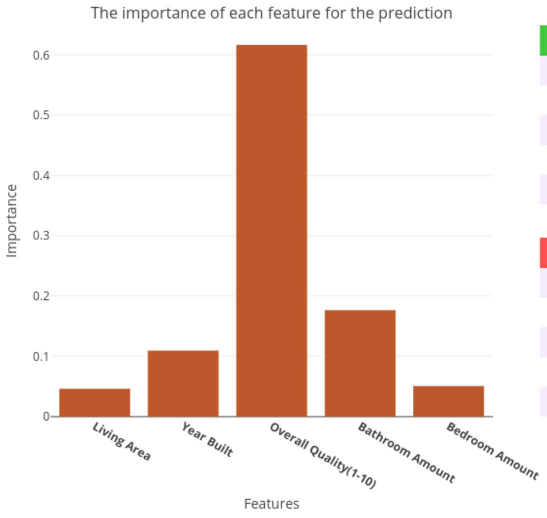
Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
113 m^2 (1218 ft^2)	2009	6	2	2
164 m^2 (1774 ft^2)	1931	7	2	2
71 m^2 (767 ft^2)	1998	7	2	1
99 m^2 (1072 ft^2)	2005	6	2	2
135 m^2 (1456 ft^2)	1978	6	2	3

Most similar houses with value High

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
244 m^2 (2633 ft^2)	2001	10	3	2
159 m^2 (1718 ft^2)	2006	10	3	3
186 m^2 (2007 ft^2)	2008	10	3	3
187 m^2 (2020 ft^2)	2009	10	3	3
219 m^2 (2364 ft^2)	2009	9	3	2

What are your likes and dislikes in the above type of explanation?

Prediction: High



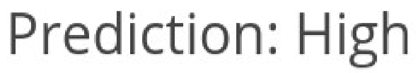
Most similar houses with value Low

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
113 m^2 (1218 ft^2)	2009	6	2	2
164 m^2 (1774 ft^2)	1931	7	2	2
71 m^2 (767 ft^2)	1998	7	2	1
99 m^2 (1072 ft^2)	2005	6	2	2
135 m^2 (1456 ft^2)	1978	6	2	3

Most similar houses with value High

Living Area	Year Built	Overall Quality(1-10)	Bathroom Amount	Bedroom Amount
244 m^2 (2633 ft^2)	2001	10	3	2
159 m^2 (1718 ft^2)	2006	10	3	3
186 m^2 (2007 ft^2)	2008	10	3	3
187 m^2 (2020 ft^2)	2009	10	3	3
219 m^2 (2364 ft^2)	2009	9	3	2

What are your likes and dislikes in the above type of explanation?



Prediction: High

What are your likes and dislikes in the above type of explanation?

Mechanical Turk Code and ID

Thank you for your participation in this survey!

Please fill in the following code in the Mechanical Turk page and provide your Mechanical Turk ID, such that we can verify that you have completed the survey.

A 5.0%	jtIrWimO7M
B 5.0%	UlYAMEC23D
C 5.0%	EbGpxhPufp
D 5.0%	KkSJ5PPivB
E 5.0%	8vnw3GkOLb
F 5.0%	0EL51CKAas
G 5.0%	IiYP4i7LPS
H 5.0%	vMgC9xKQk8
I 5.0%	QA6axRlt7j
J 5.0%	mIy3kS3ZUj
K 5.0%	sKUoigIIOo
L 5.0%	4gT9B7WxXc
M 5.0%	VTTtWpgft5
N 5.0%	ngoxFeqUue
O 5.0%	9MxozFmZb1
P 5.0%	6vPkJm0ter
Q 5.0%	yjJmCxL0Iy
R 5.0%	HrFUNHoXHE
S 5.0%	C56EOGHsq5
T 5.0%	ZRXAQY7DU4

What is your worker ID of mechanical turk?

If you have remarks on the survey, please feel free to write them below.
Have a nice day further and many thanks :)