Diploma Thesis

EVALUATION OF EXPLAINABILITY TOOLS

Alexis Eftychiou

UNIVERSITY OF CYPRUS

DEPARTMENT OF COMPUTER SCIENCE

May 2022

UNIVERSITY OF CYPRUS

DEPARTMENT OF COMPUTER SCIENCE

Evaluation of Explainability Tools

Alexis Eftychiou

Supervisor
George Papadopoulos

The Diploma Thesis was submitted for partial fulfilment of the requirements for obtaining the degree of Computer Science of the Department of Computer Science of the University of Cyprus

May 2022

Acknowledgements

First of all, I would like to thank my supervisor, Professor George Papadopoulos for giving me the opportunity to work on this project.

Furthermore, I would like to thank Styliani Kleanthous and Kalia Orphanou for the assistance they provided to me through meetings and for the feedback that helped me immensely.

Finally, I would like to thank my family and friends for all the support they provided me.

Abstract

The thesis is focused on Explainable Artificial Intelligence (XAI), where we will talk Machine Learning (ML) and Deep Learning (DL) alongside it. The aim of thesis is to understand the terms mentioned by giving their definition, analysing them and explaining the connections they have between them. After that, there is the introduction of the model agnostic methods, LIME and SHAP, both of which are explained by defining them and giving a few examples. At the end there is a comparison of the two methods to showcase their strengths and weaknesses.

Later on, there is a methodology chapter that focuses on what our experiment will be. It talks about the dataset that is used, the ‘Student Performance Data Set’[31] and breaks down the features and data entries. There is also talk of the Random Forest Regressor that is used to calculate the prediction of a student’s performance from the dataset. After this, the experiment takes place where LIME and SHAP are put in to use to provide explanations for the predictions made for two different user inputs. In this chapter there are graphs and plots, that try to show dependencies and the importance that each feature has on the final output.

In the end, there is talk of the results of the experiment in order to understand them. There is discussion of what was achieved in the thesis, a summary of everything that was discussed and talk about the future of XAI.

Content

Chapter 1

Introduction

1.1 Motivation

The first thing that I wanted to answer when I took on this subject, was to satisfy my curiosity and find out why all these Artificial Intelligence (AI) systems I kept hearing about made the choices that they made. I wanted to see how pure data could help a Machine Learning (ML) model be trained, in order for it to provide outcomes. These outcomes need explaining and my motivation was to learn how to create such explanations through this thesis.

1.2 Aim of the thesis

The aim of in this thesis was to first understand the concept of Explainable Artificial Intelligence (XAI). Through the research I have done, I discovered that even the most well-known explainability methods, LIME and SHAP, have still a long way to go before they can truly be adopted as fully trustworthy models. Firstly, this thesis aimed at learning about these two methods and then though my own experiments this thesis aimed at exploring how the abovementioned methods create their explanations, prove dependencies between features and calculate importance of these features. Using the dataset mentioned in Chapter 4, which focuses on predicting the grades of high school students, I wanted to see if the collected data can actually make a solid prediction.

1.3 Structure

Firstly, Chapter 2 is focused on explaining ML and its three categories, supervised ML, unsupervised ML, and reinforcement learning. We take a look at Deep Learning, Artificial Neural Networks, black box models and machine learning in education. Closing Chapter 2, XAI is explained, focusing on explainability and Interpretability, and the explainability approaches.

Chapter 3 is all about the model agnostic methos. First for LIME, there is a definition, some examples for text classification, tabular data and image data, and a list of pros and cons. For SHAP, there is a definition, some examples for tree ensemble, DL with gradient explainer and single instance text plot, and a list of pros and cons.

In Chapter 4, we take a look at the Methodology of the experiments that will follow in Chapter 5. We talk about the purpose of the experiment, analysed the dataset that is used and talked about the datasets features, discussed the classification model that was chosen and showed LIME's and SHAP's implementation.

Chapter 5 is all about running the experiment, by producing graphs and plots for two different user inputs. There are graphs about importance and dependence and then graphs about specific grade prediction explanation.

Closing in Chapter 6, we discuss the results in Chapter 5, give a summary of the thesis and talk about what we have learned and talk about further future work that can be done in the field of XAI.

Chapter 2

Explainable Artificial Intelligence

2.1 Machine Learning

So, what is ML and what is its relation to AI? A pioneer of the field of ML, Arthur Samuel defined ML as a *“Field of study that gives computers the ability to learn without being explicitly programmed”*[21]. ML algorithms will use concentrated data as input to produce a prediction as an output. ML is widely used by big enterprises in order for them to understand customer behaviour and adapt to their expectations. Big companies like Google, Microsoft and Facebook will use the data that they get from their customers to provide them with a better experience but more importantly to them, to find ways to create more targeted features and advertisements to monetize the customers more efficiently.[23] ML is not only used by commercial enterprises but also in other aspects in society. It can be very useful in medicine where it can help patients get quicker diagnosis of potential illnesses.[22] It can also help in education, like the model presented in this thesis, in order to spot strengths and weakness of students and help them pre-emptively and prevent potential failures. The amount of data we have increases rapidly and so are the worries of how it is used. Questions of who has access to it and who will actually reap the benefits will have to be always answered in order to be kept in check. In the next sections we will see the three biggest ML methods (Figure 2.1) and what each of them offers.

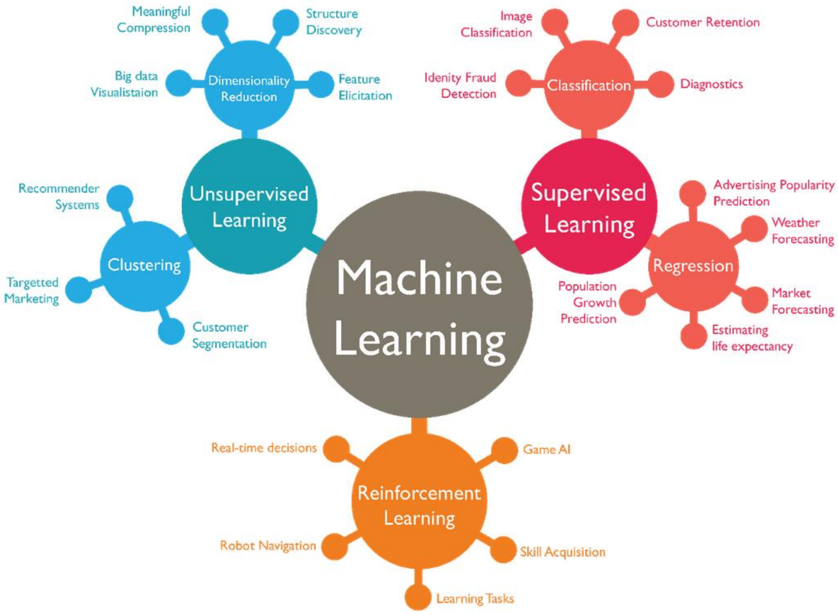


Figure 2.1: The ML methods.[19]

2.2.1 Supervised Machine Learning

The most popular ML category is Supervised Machine Learning [2] which is of course based on supervision. With this method we train the machine using labelled datasets and based on the training that we give it, the machine will predict the output. Some supervised learning systems, include spam classifiers of email, face recognizers over images, medical diagnosis systems for patients and speech recognition.

Supervised Machine Learning follows these 7 steps:

1. The collection of data. Here is where the choice of the dataset is made. It is important to make sure that the data is reliable so that correct patterns can be found. The dataset in the case of tabular data, contains rows which are the sample and columns that are the features.

2. The preparation of data. Here the data is split into two sets, a training set, which is the set your model learns from, and a testing set, which is used to check the accuracy of your model after training.
3. Choosing a model. Here you choose a ML model that is relevant for the kind of data you are working on.
4. Training the model. This is the most important step in ML, where you pass on the data to your ML model so it can find patterns and make predictions.
5. Evaluating the model. The testing data from step 2 is used here to get an accurate measure of how your model will perform.
6. Parameter Tuning. Once everything is done, you can tune the parameters that are present in your model, so you can see if the accuracy can be improved in some way.
7. Making Predictions. Finally, you can use your model on data from the testing set to make predictions accurately.

The two categories of Supervised Machine Learning are Classification and Regression. Classification algorithms are used to solve problems where the output is always categorical. For example, whether someone is male or female, a flower is green or yellow and simple “yes” or “no” questions. Regression algorithms on the other hand are for problems where there is a linear relationship between the input and output variables. This algorithm is helpful for predicting continuous output variables for such thing as weather predictions and the stock market.

2.2.2 Unsupervised Machine Learning

We also have Unsupervised Machine Learning [2] where obviously there is no need for supervision. More broadly, unsupervised learning involves the analysis of unlabeled data which the machine will use to predict the output without any sort of supervision. This method is widely used for recommendation systems, which help build applications for used such as e-commerce websites, but also network analysis which can help identify plagiarism and copyright infringements.

One of the categories of Unsupervised learning is Clustering, which is used to find inherent groups in our datasets and group them into a cluster in such a way that the objects with the most similar traits remain in one group and have very little to nonsimilar traits with objects from other groups. Some popular clustering algorthms are the K-Means Clustering algorithm and the Mean-shift algorithm. The other category is Association whose purpose is to find interesting relations between variables in a dataset, more precisely the dependency of one data item to another different data item. This algorithm can be used for such applications as web usage mining and continuous production.

2.2.3 Reinforcement Learning

The third major ML method is Reinforcement Learning [3]. In this method the information available in the training data is intermediate between supervised and unsupervised learning. Here, we do not have labelled data like we do in supervised learning and the only way for an AI agent to learn is from their own experiences. What this means is that the training data can only provide an indication as to whether the action performed is correct and in the case is not the problem of finding the correct action still remains.

The two main categories of Reinforcement Learning are firstly the Positive Reinforcement Learning which increases the tendency that the required behaviour would occur again by enhancing the strength of the behaviour of the agent in a positive manner and the Negative Reinforcement Learning which increase the tendency of the specific behaviour would occur once again by way of avoiding the negative condition. This method is used a lot in the video game and robotics sections where it helps AI behaviour to get better.

2.3 Deep Learning

In 2006 Hinton et al. introduced deep learning (DL), which was based on the concept of artificial neural networks (ANN), which function by feeding data to a model and letting figure out itself whether it made the right decision or interpretation about a given data element.[7] DL is a type of ML that tries to imitate the way humans gain knowledge. It is a very important part of data science, which encompasses statistics and predictive modelling. It is extremely helpful for data scientists whose task is to analyse and interpret massive amounts of data, which is a thing DL can help make much easier and faster.

2.3.1 Artificial Neural Networks

There are different kinds of ANNs such as the Multi-Layer Perceptron, the Convolutional Neural Network and the Recurrent Neural Network. The MLP, a supervised learning approach, is a type of feedforward ANN. A typical MLP is a fully connected network that consists of an input layer that receives input data, an output layer that makes decisions or predictions about the input signal, and one or more hidden layers between these two that are considered as the network’s computational engine. The CNN is a popular discriminative deep learning architecture that learns directly from the input without the need for human feature extraction. Each layer in CNN considers the optimum parameters for a meaningful output as well as reduces model complexity. A RNN employs sequential or time-series data and feeds the output from the previous step as input to the current stage. Like CNN, recurrent networks learn from training input, however, distinguish by their ‘memory’, which allows them to impact current input and output through using information from previous inputs.[7]

2.4 Black Box Models

The term “black box” is shorthand for models that are sufficiently complex that they are not straightforwardly interpretable to humans. Black box models are also known as opaque models. Lack of interpretability in predictive models can under-mine trust in those models, especially in health care, in which so many decisions are life and death issues. No model is perfect, so it is entirely reasonable that a doctor would be weary of blindly trusting the prediction of a model that cannot provide any insight to its decision.[12] The inability of obtaining an explanation for what one considers a biased decision is a profound drawback of learning from big data, limiting social acceptance and trust on its adoption in many sensitive contexts.[6] Which is why explainability techniques are used to provide the needed transparency to the processing of the algorithm and the final outcome. It is important to note that DL and neural networks are also considered black box models, since for neural networks we do not know how all the individual neurons work together to arrive at the final output.

2.5 Machine Learning in Education

ML can be used in many ways to help improve the education sector. It can predict student performance so it can help find the weakness in students and provide them with the necessary help. It can also grade students fairly by providing feedback to both the teachers and students about how the student learns. Improving student retention can also be achieved by identifying “at risk” students pre-emptively so the school can detect them and help them. It can even support teachers by applying ML models on the students handwritten papers to classify/predict their grade.[9]

The most essential of these problems that can be solved with ML is the prediction of student performance with the help of big datasets. The data used could vary from smaller things such as the size of the family and the parents’ occupation to much more useful and important data such as study time and internet access and finally most importantly the student’s previous grades. By having this data from students, we can discover how important each piece of data is to them and how much importance does it have on their predicted grade. To try and solve this problem we seek to improve the effectiveness of the education process through the recognition of patterns in the students’ performance and general life

2.6 Explainable Artificial Intelligence

According to the Oxford Languages, the word’s “explain” definition is “to make (an idea or situation) clear to someone by describing it in more detail or revealing relevant facts”. The idea in this scenario is for Artificial Intelligence (AI) and the results of AI algorithms to be described and reveal the reasoning for those results. This is known as Explainable Artificial Intelligence (XAI). XAI strives to turn the questions we have as to why a Machine Learning (ML) system which is fed huge amount of data, makes the decisions it makes (Figure 2.2).

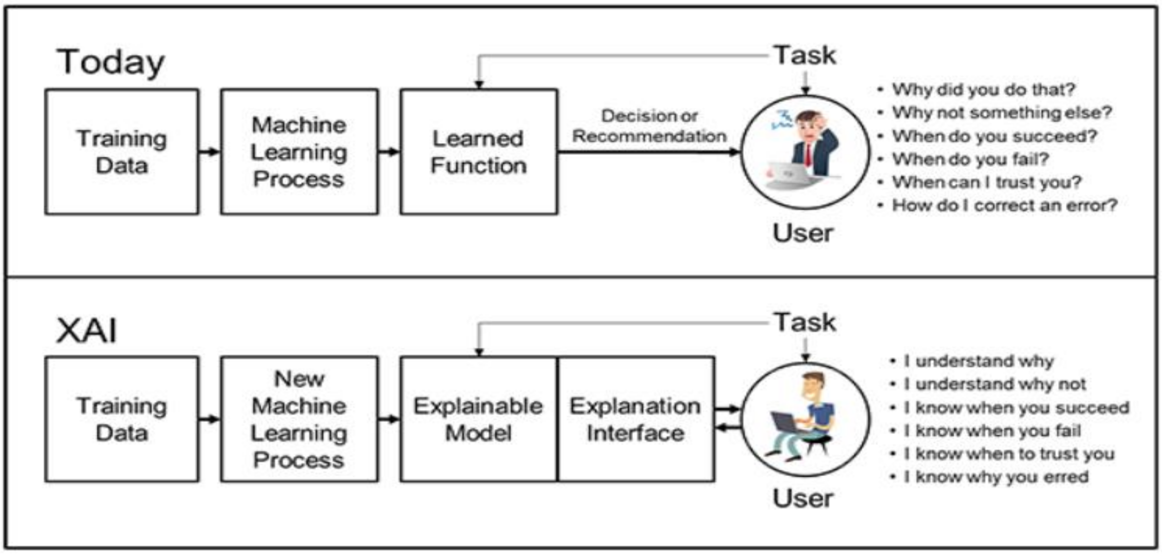


Figure 2.2: The concept of XAI.[20]

From the moments scientist started research on AI it was important to them that these intelligent systems could explain their AI results, especially when it comes to decisions.[4] For example, if a rule-based system rejected your credit card payment, it should always provide an explanation and reasoning for the negative decision. Back then though, AI algorithms were much easier to interpret but today even the most straightforward deep neural network cannot be easily understood. This is where Explainable AI comes into place. XAI algorithms are used to efficiently explain decisions made by AI to their specific target audience (Figure 2.3). If the AI makes its own decisions, for example braking of the car, selling shares, issuing a traffic punishment order, the affected people must be able to understand the reason. Explainable AI helps developers to improve AI algorithms, by detecting data bias, discovering mistakes in the models, and remedying their weaknesses.

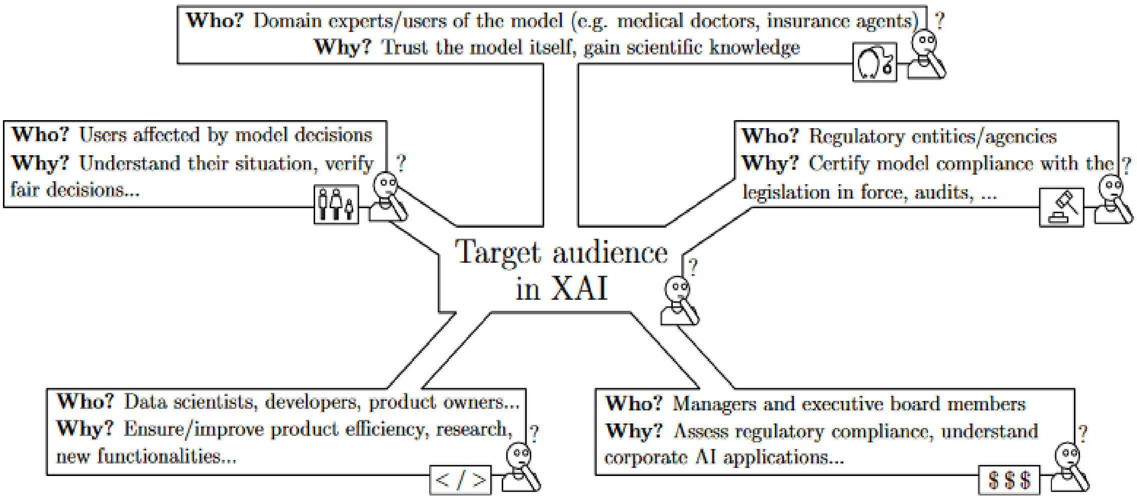

Who? Domain experts/users of the model (e.g. medical doctors, insurance agents)
Why? Trust the model itself, gain scientific knowledge
Who? Users affected by model decisions
Why? Understand their situation, verify fair decisions...
Who? Regulatory entities/agencies
Why? Certify model compliance with the legislation in force, audits, ...
Target audience in XAI
Who? Data scientists, developers, product owners...
Why? Ensure/improve product efficiency, research, new functionalities...
&lt; / &gt;
Who? Managers and executive board members
Why? Assess regulatory compliance, understand corporate AI applications...
$ $ $
?

Figure 2.3: Diagram showing the different purposes of explainability in ML models sought by different audience profiles.[11]

2.6.1 Explainability and Interpretability

Explainability is often interchangeably misused with Interpretability but there are notable differences between the two concepts. For a system to truly thrive, both of them must be taken into consideration. To get a clearer idea of what each of them means, in this section we will clarify what they actually are in relation to XAI.[11]

Explainability: is associated with the notion of explanation as an interface between humans and a decision maker that is, at the same time, both an accurate proxy of the decision maker and comprehensible to humans. Explainability is the ability to explain quite literally what is happening.

Interpretability: it is defined as the ability to explain or to provide the meaning in understandable terms to a human. In other words, interpretability is about being able to discern the mechanics without necessarily knowing why.

2.6.2 Explainability Approaches

Explainability techniques can be applied both in Transparent ML Models and Opaque ML models. A model is considered to be transparent if, by itself, it has the potential to be understandable. In other words, transparency is the opposite of a “black-box”.[24] Typical transparent models include k-nearest neighbours, decision trees, rule-based learning, Bayesian network and others. For transparent models, we mostly use visualization methods to understand the dataset.

Opaque models are more 'black box' in nature. Although these models often achieve high accuracy, they are not transparent. Explainability techniques are mostly applied to Opaque models since their opaqueness needs to be presented in an understandable way. For Opaque ML models, post-hoc explainability techniques are used which can be classified are either model-agnostic or model-specific Model-agnostic XAI approaches are designed with the purpose of being generally applicable. As a result, they have to be flexible enough, so that they do not depend on the intrinsic architecture of the model, thus, operating solely on the basis of relating the input of a model to its outputs. Model-specific XAI approaches often take advantage of knowing a specific model and aim to bring transparency to a particular type of one or several models. Typical opaque models include random forest, neural networks, SVMs, and so on.[3]

Some of the most popular categories of post-hoc explainability techniques are Explanation by simplification, Explanation by feature relevance, Visual explanation, and Local explanation.

Explanation by simplification: By simplifying a model via approximation, we can find alternatives to the original models to explain the prediction we are interested in. This new, simplified model usually attempts at optimizing its resemblance to its antecedent functioning, while reducing its complexity, and keeping a similar performance score.[3]

Explanation by feature relevance: This idea is similar to simplification. Roughly, this type of XAI approaches attempts to evaluate a feature based on its average expected marginal contribution to the model's decision, after all possible combinations have been considered.[3]

Visual explanation: This type of XAI approach is based on visualization. As such, the family of data visualization approaches can be exploited to interpret the prediction or decision over the input data.[3]

Local explanation: Local explanations approximate the model in a narrow area, around a specific instance of interest, and offer information about how the model operates when encountering inputs that are similar to the model we are interested in explaining.[3]

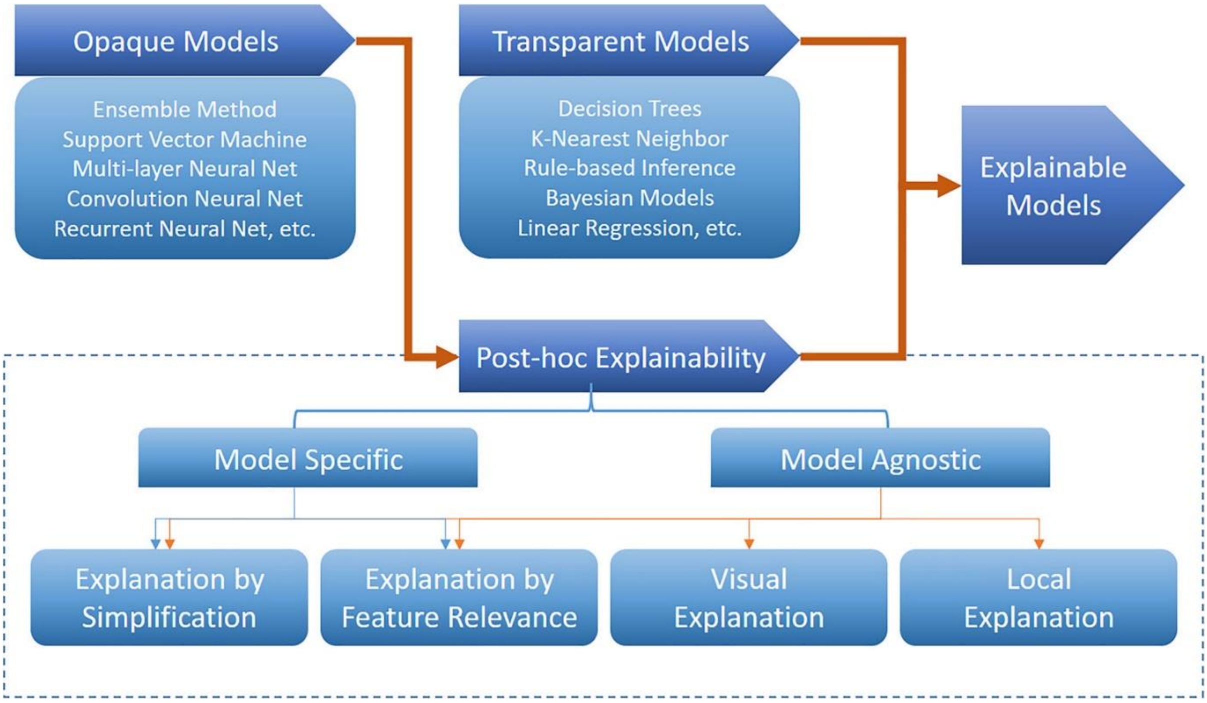


Figure 2.4: The high-level ontology of explainable artificial intelligence approaches. It displays the classification of explainability techniques.[11]

Chapter 3

Model Agnostic Methods

Model Agnostic methods, can explain any ML model regardless if the model itself is interpretable or not. As we mentioned above, these methods are called post-hoc, which means the explanation comes after the training of the model using outside methods. These post hoc models will help keep high accuracies in complicated models and at the same time keep the interpretability at its possible best. There are a few characteristics that we want our methods to have.

3.1 LIME - Local Interpretable Model-Agnostic Explanations

The LIME method, which was originally proposed by Ribeiro, Singh, and Guestrin in 2016[14], is a technique that approximates any black box machine learning model with a local, interpretable model to explain each individual prediction.

3.1.1 Definition

In order for us to be model agnostic, we have to perturb the input and see how the predictions change. This will help us with interpretability because we can perturb the input by changing some components that make more sense to humans, like words or parts of images, even if the model is using more complicated components as features, for example word embeddings for text classification.[13]

LIME takes into consideration the set of features that the ML model will be applied on and computes the weight of each feature on the classification outcome. Based on what that outcome is, it will display the features and their weight in a way that explains the outcome. Tabular data is data that comes in tables, where each row represents a sample and each column a feature. For a text classifier the features are words or phrases from the vocabulary that are mapped to vectors of real numbers.

In Figure 3.1 the process of explaining individual predictions is illustrated. A doctor will be much better positioned to decide with the help of a model if the intelligible explanations needed are adequately provided. For this specific case, an explanation is a small list of symptoms with relative weights – symptoms that either contribute to the prediction (showed in green) or are evidence against it (showed in red). With the prior knowledge that humans possess about the application domain, they can choose themselves whether to accept or reject a prediction if they understand the reasoning behind it.[14]

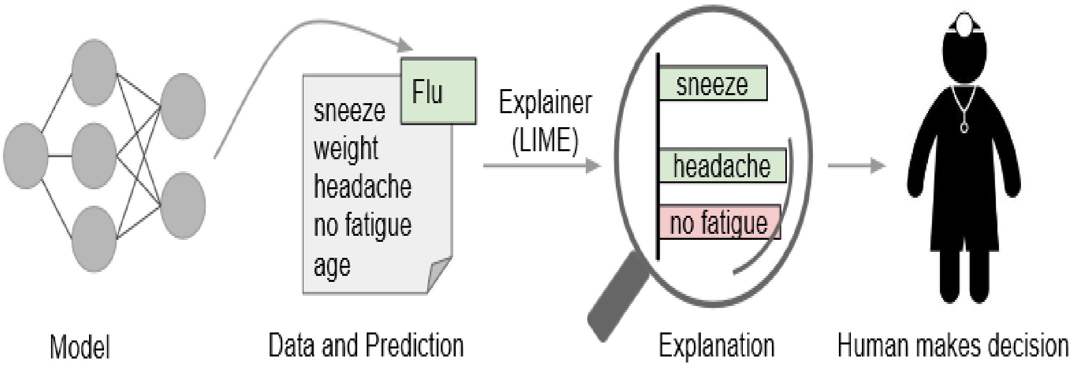


Figure 3.1 Explaining individual predictions. A model predicts that a patient has the flu, and LIME highlights the symptoms in the patient's history that led to this prediction. Sneeze and headache are portrayed as contributing to the "flu" prediction, while "no fatigue" is evidence against it. With these, a doctor can make an informed decision about whether to trust the model's prediction [14]

3.1.2 Example - Text Classification for a Multiclass Case

This first example in Figure 3.2 represents the results from a multi-class text classification. The dataset used is ‘The 20 Newsgroups data set’.[26] Here the features are words taken from a number of papers and the goal was to classify the topic of the paper considering five classes: atheism, Christianity, religion. misc, Mideast and other. The fact that the five classes have many common words making them hard to distinguish.

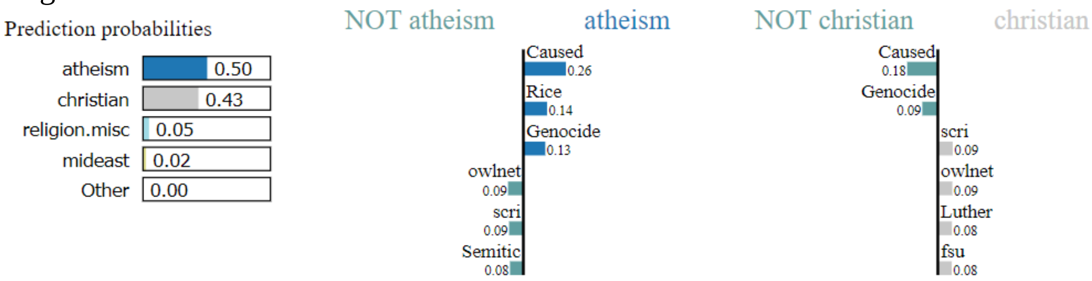
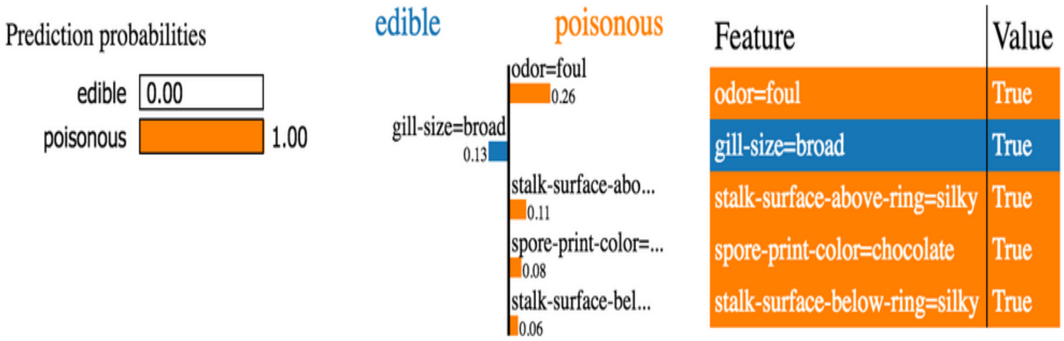


Figure 3.2 Explanation for a prediction for a multiclass case [25]

The LIME method on the left of Figure 4.2 predicts that there is a 50% chance that the paper that this example takes the words from, refers to atheism. There is also 43% chance the topic of the paper is the Christian religion, and the other classes have very small chances. The other part of Figure 4.2 shows us the 6 most impactful words, whether that is positive or negative. Words such as ‘caused’, ‘rice’ and ‘genocide’ have a positive influence on the atheism prediction, which classify the topic of the paper to be related to atheism, but on the other hand words such as ‘owlnet’, ‘scri’ and ‘semitic’ have a negative influence on the atheism prediction, which classify the topic of the paper to be related to the Christian religion. The impact on the Christian prediction is seen on the far-right side of Figure 4.2.

3.1.3 Example - Tabular Data

In the next example tabular data are used to determine if mushroom is ‘edible’ or ‘poisonous’. The dataset used is ‘Mushroom Data Set’.[27] In Figure 3.3 each feature is given a value, either true or false, which will give show the influence each of them will have on the final prediction.



Feature	Value
odor=foul	True
gill-size=broad	True
stalk-surface-above-ring=silky	True
spore-print-color=chocolate	True
stalk-surface-below-ring=silky	True

Figure 3.3 Explanation for a prediction for tabular data [25]

The prediction that was made, shows that the mushroom is poisonous. Most of the influential features are positive towards the fact that is poisonous. Things like the ‘odour’, ‘the stalk surface above ring’ and the ‘spore print colour’ have a positive influence, while the ‘grill size’ has a negative influence.

3.1.4 Example – Image

In this section with LIME, explanations are generated for image classification tasks.[28] The basic idea is to understand why a ML model predicts that an instance (image) belongs to a certain class (labrador in this case). The picture in Figure 3.4 is used to predict the breed of the dog.

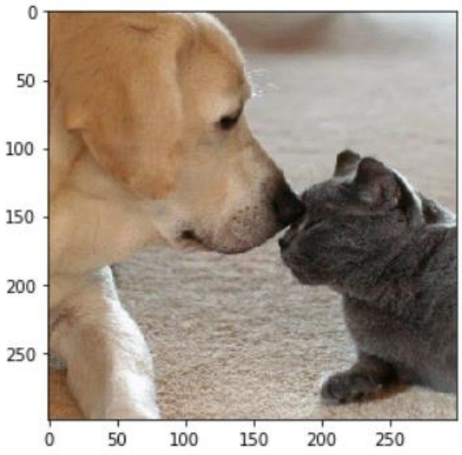


Figure 3.4 Unaltered Original Image [28]

The first step is generating perturbations by turning on and off some of the superpixels in the image. Using the quickshift segmentation algorithm, pixels are generated (68 superpixels in total). The generated superpixels are shown in Figure 3.5.

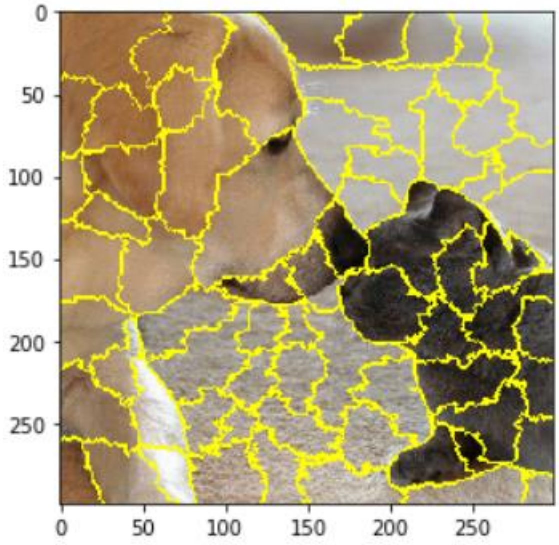


Figure 3.5 Generated superpixels on image [28]

Creating random perturbations (150 in this example), the following image in Figure 3.6 is created.

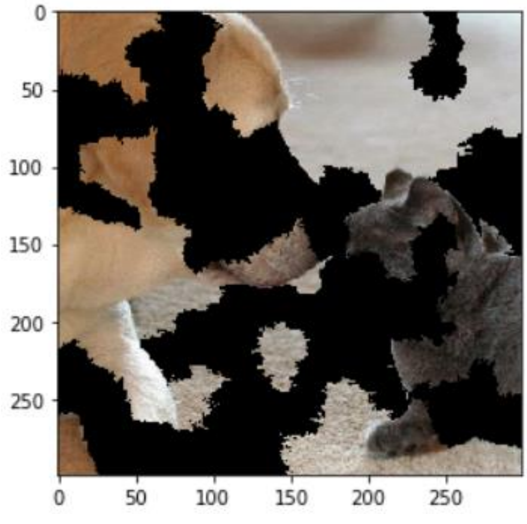


Figure 3.6 Perturbed Image [28]

Using Inception V3, which is a convolutional neural network, it predicts classes of new generated images, computes distances between the original image and each of the perturbed images and uses the kernel function to compute weights. Finally using the perturbations, predictions and weights, we will compute the features (superpixels) and get our final image (Figure 3.7) which is the LIME explanation as to why the breed of the dog is a labrador

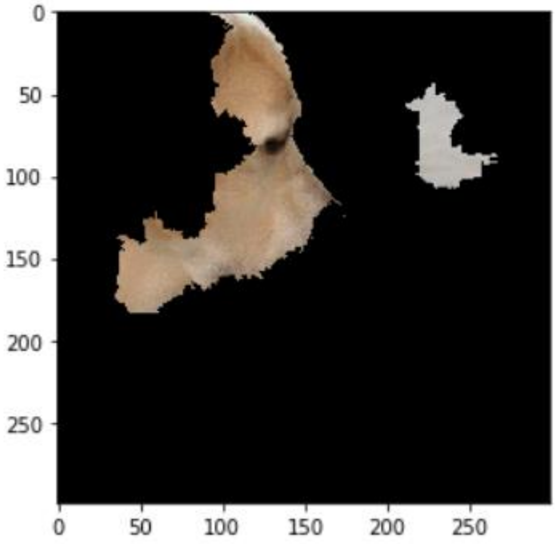


Figure 3.7 Final image which explains the prediction [28]

3.2 SHAP - SHapley Additive exPlanations

SHapley Additive exPlanations (SHAP) was first introduced by Lundberg and Lee in 2017 [16] and is a game theoretic approach to explain the output of any machine learning model. It connects optimal credit allocation with local explanations using the classic Shapley values from game theory and their related extensions.

3.2.1 Definition

SHAPs goal is to explain the prediction of an instance x by computing the contribution of each feature to the prediction. The SHAP explanation method computes Shapley values from coalitional game theory. The feature values of a data instance act as players in a coalition. Shapley values tell us how to fairly distribute the “payout” among the features. A player can be an individual feature value, as in the tabular datasets or a group of feature values i.e. word embedding vectors in the text data. An example is in order to explain an image, pixels can be grouped to superpixels, and the prediction distributed among them. One innovation that SHAP brings to the table is that the Shapley value explanation is represented as an additive feature attribution method, a linear model. That view connects LIME and Shapley values.

3.2.2 Example – Tree Ensemble

While SHAP can explain the output of any ML model, a high-speed exact algorithm for tree ensemble methods has been developed in [17], that is supported by tree models such as xgboost, applying to the Boston Housing Dataset.[29]

Firstly, there is the Waterfall plot as seen Figure 3.8 which shows features each contributing to push the model output from the base to the model output. Features pushing the prediction higher are in red and those pushing the prediction lower are in blue. Stats like LSTAT (% lower status of the population), PRATIO (pupil-teacher ratio by town) and AGE(proportion of owner-occupied units built prior to 1940) push the prediction higher while RM (average number of rooms per dwelling), NOX (nitric oxides concentration) and RAD (index of accessibility to radial highways) push the prediction lower.

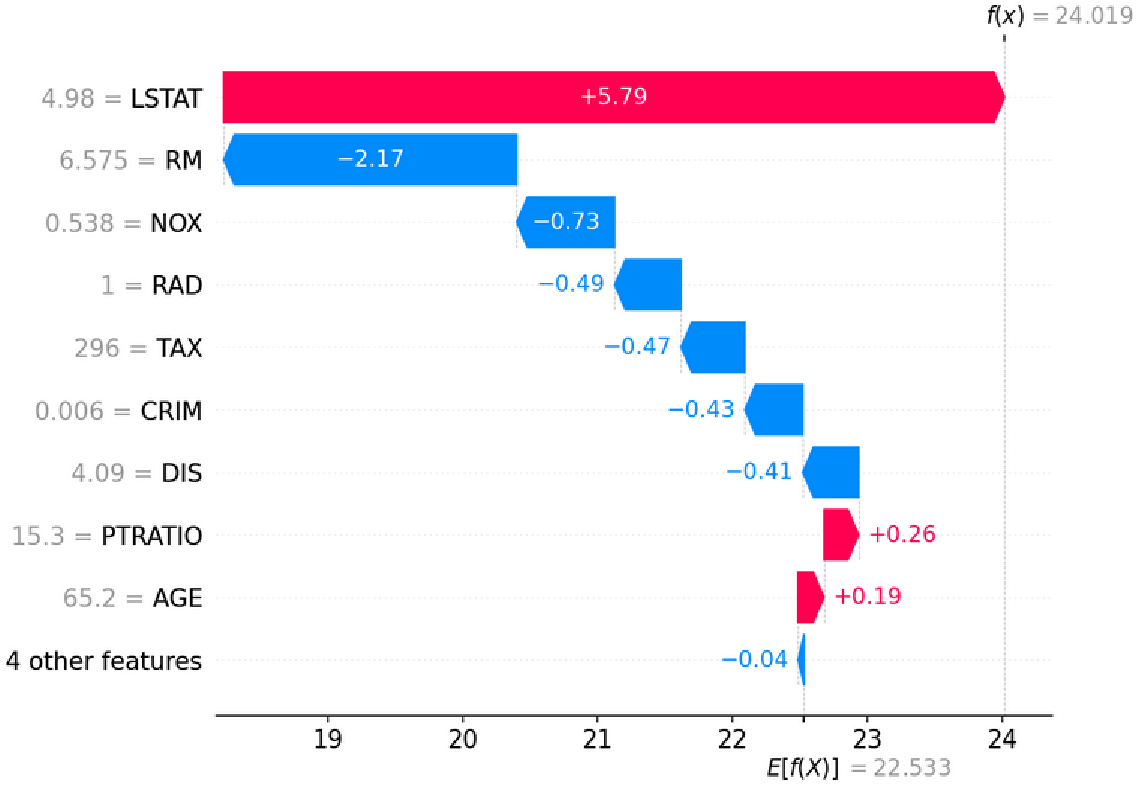


Figure 3.8 The Waterfall plot showing each features contribution to the model output.[17]

We also have the Force Plot (Figure 3.9) that show the same information as the Waterfall plot but in a different way

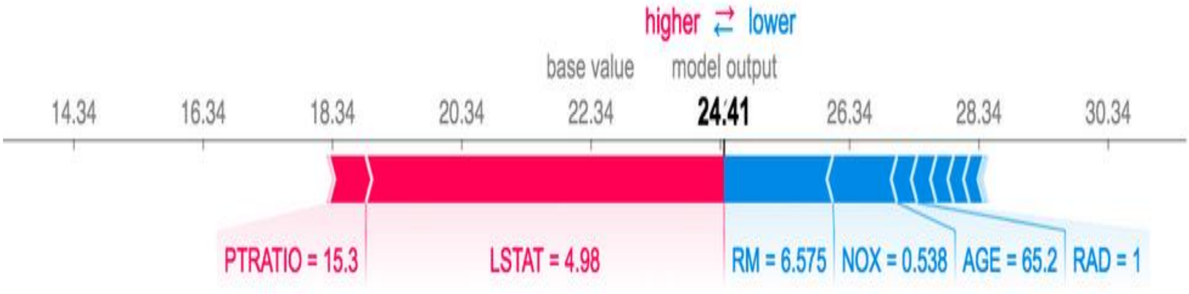


Figure 3.9 The Force Plot [17]

If we take many force plot explanations such as the one shown in Figure 4.9, rotate them 90 degrees, and then stack them horizontally, we can see explanations for an entire dataset in Figure 3.10.

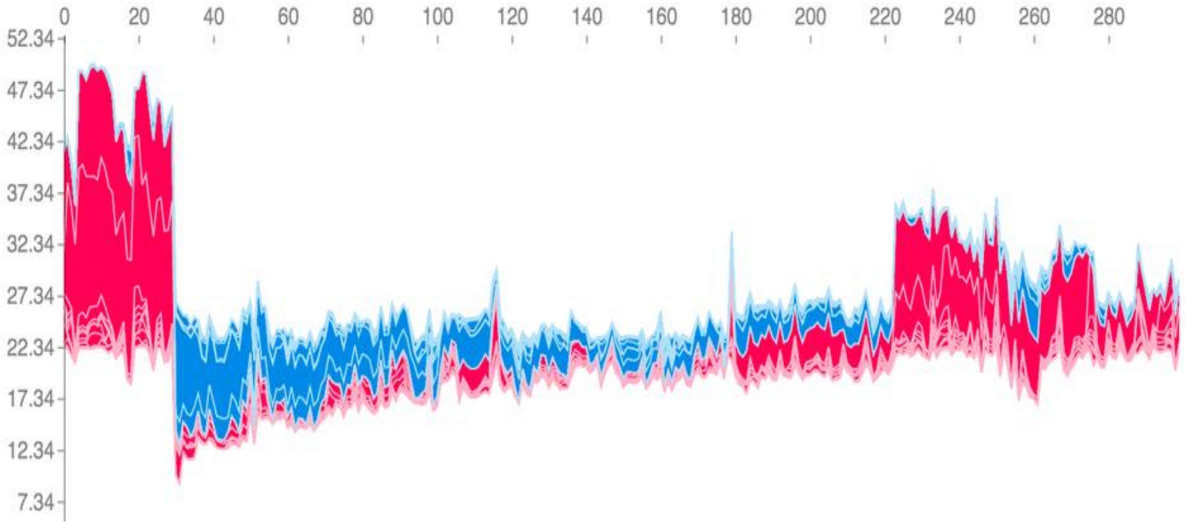


Figure 3.10 The Force Plot for the entire dataset [17]

To understand how a single feature effects the output of the model we can plot the SHAP value of that feature vs. the value of the feature for all the examples in a dataset (Figure 3.11). Since SHAP values represent a feature's responsibility for a change in the model output, the plot below represents the change in predicted house price as RM changes. If we pass the whole explanation tensor to the colour argument the scatter plot will pick the best feature to colour by. In this case it picks RAD since that highlights that the average number of rooms per house has less impact on home price for areas with a high RAD value.

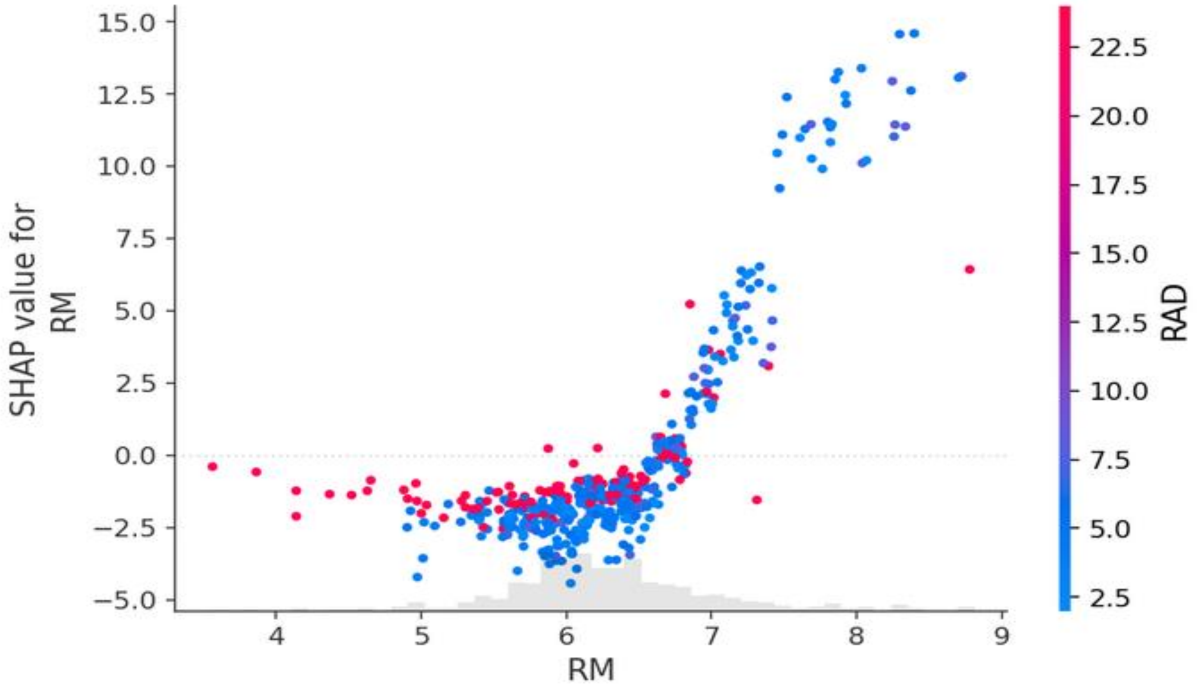


Figure 3.11 The Dependence Plot [17]

To get an overview of which features are most important for a model we can plot the SHAP values of every feature for every sample. The plot below (Figure 3.12) sorts features by the sum of SHAP value magnitudes over all samples and uses SHAP values to show the distribution of the impacts each feature has on the model output. The colour represents the feature value (red high, blue low). This reveals for example that a high LSTAT lowers the predicted home price, a high RM increases the predicted home price and low TAX lowers the predicted house value.

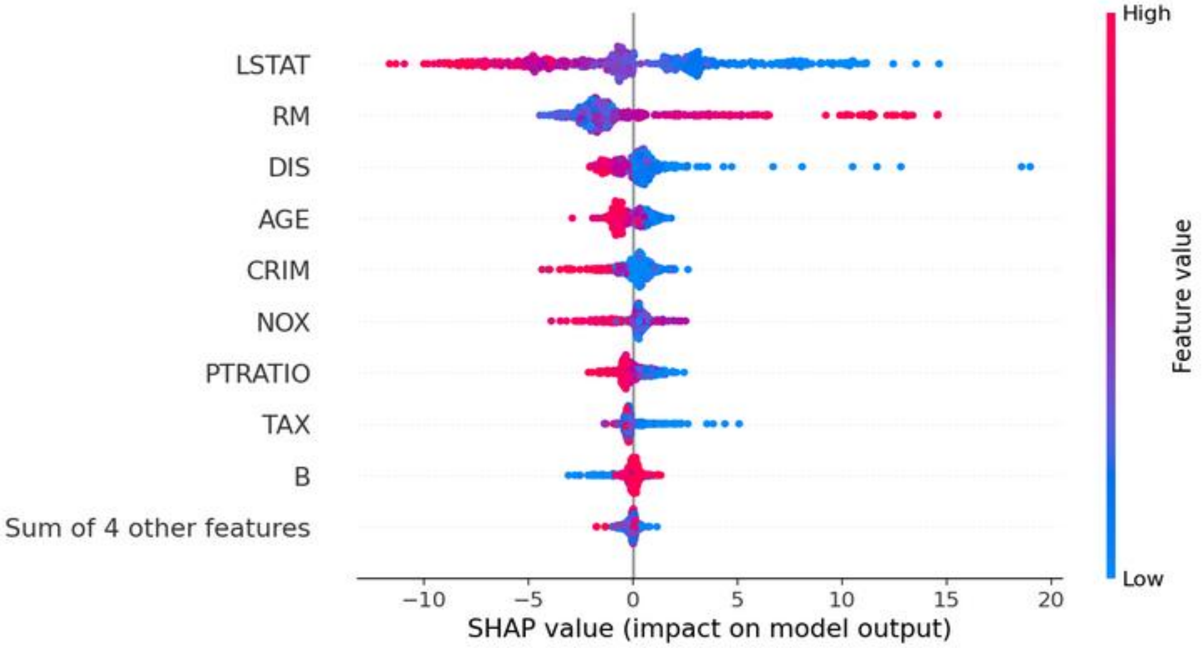


Figure 3.12 The Summary Plot [17]

3.2.3 Example – Deep Learning with Gradient Explainer

Gradient Explainer allows an entire dataset to be used as the background distribution (as opposed to a single reference value). If we approximate the model with a linear function between each background data sample and the current input to be explained, and we assume the input features are independent then expected gradients will compute approximate SHAP values. Predictions for two input images are explained in Figure 3.13. Red pixels represent positive SHAP values that increase the probability of the class, while blue pixels represent negative SHAP values the reduce the probability of the class [17].

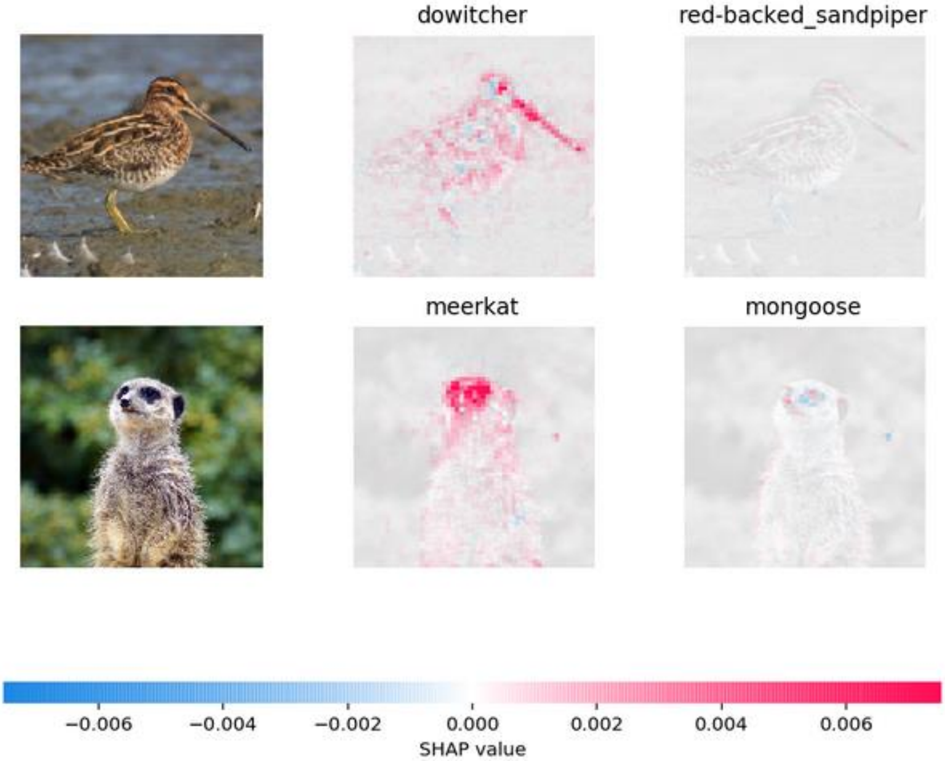


Figure 3.13 Deep Learning with Gradient Explainer

3.2.4 Example – Single Instance Text Plot

When we pass a single instance to the text plot, we get the importance of each token overlayed on the original text that corresponds to that token. Red regions correspond to parts of the text that increase the output of the model when they are included, while blue regions decrease the output of the model when they are included. In the context of the sentiment analysis model here red corresponds to a more positive review of the movie and blue a more negative review. Figure 3.14 shows the impact of each word in a review of a movie. Here, the Hugging Face transformers library is used.[30]

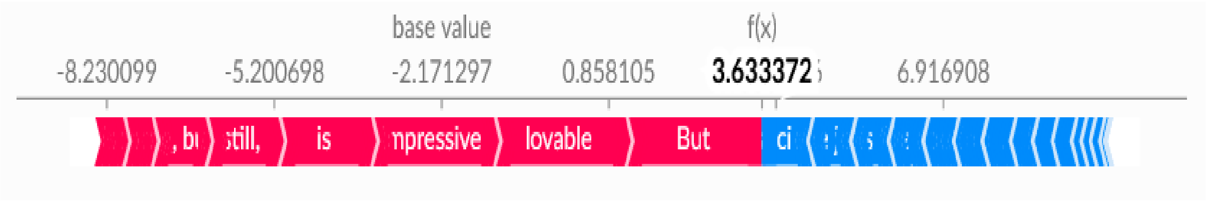


Figure 3.14 Single Instance Text Plot [17]

3.3 Comparison

Points of comparison between LIME and SHAP:

Model Agnostic : both explainability methods are model agnostic, as they do not imply any assumptions about the black-box model structure.

Speed: they both have fast implementation for tree-based models, but LIME is generally faster than shapley values which take longer to compute.

Interpretation: both of them can create intentionally misleading interpretations, which can hide biases.[16]

Classifiers: both can use different types of classifiers like text, image, tabular, etc.

Explainability: both explore and use the property of local explainability to build surrogate models to black-box ML models to provide them interpretability.

Consistency: LIME has gotten criticism over the lack of stability, consistency and missingness, three properties that are fulfilled by SHAP.[16]

Guarantees: due to its theoretical guarantees and simplicity, SHAP is widely used and maybe more acceptable.[17]

Chapter 4

Methodology

4.1 Purpose

The main goal of the studies that follow in Chapter 5, was to get a first-hand experience on how the entire process happens. From choosing a dataset all the way to getting the predicted output. I wanted to see the results and explanations given by LIME and SHAP, using test data that I input myself. I wanted to understand how some features that are not as obvious as previous semester grades, could actually affect a student's performance. How much do absences from class or even a student's family size truly has an impact on performance? This is what I try to answer in Chapter 5.

4.2 Dataset

The dataset used in this experiment is the 'Student Performance Data Set'.[31] The description of the dataset is as follows "This data approach student achievement in secondary education of two Portuguese schools. The data attributes include student grades, demographic, social and school related features) and it was collected by using school reports and questionnaires. Two datasets are provided regarding the performance in two distinct subjects: Mathematics (mat) and Portuguese language (por). Important note: the target attribute G3 has a strong correlation with attributes G2 and G1. This occurs because G3 is the final year grade (issued at the 3rd period), while G1 and G2 correspond to the 1st and 2nd period grades."

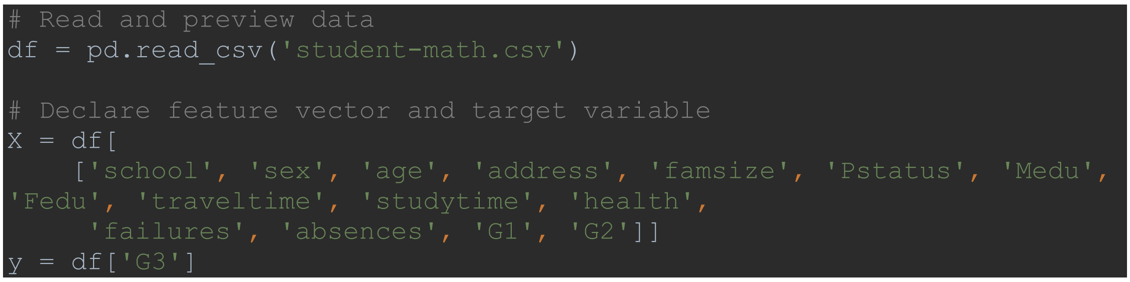
For the purpose of this experiment, I used 15 features from the dataset that a user can input (excluding the choice between LIME and SHAP, and the course dataset) and the class variable. I chose 15 out of the available 32 input features, in order to keep features that have a big influence on the result and to have graphs that do not become messy from having too many features. The features that I chose are as follows (the numbers are what the user chooses, to correspond to their choice):

1. school - student's school (binary: "0" - Gabriel Pereira or "1" - Mousinho da Silveira)
2. sex - student's sex (binary: "0" - male or "1" - female)
3. age - student's age (numeric: from 15 to 22)
4. address - student's home area type (binary: "0" - urban or "1" - rural)
5. famsize - family size (binary: "0" - less or equal to 3 family members or "1" - greater than 3 family members)
6. Pstatus - parent's cohabitation status (binary: "0" - living together or "1" - apart)
7. Medu - mother's education (numeric: 0 - none, 1 - primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education)
8. Fedu - father's education (numeric: 0 - none, 1 - primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education)
9. traveltime - home to school travel time (numeric: 1 - <15 min., 2 - 15 to 30 min., 3 - 30 min. to 1 hour, or 4 - >1 hour)
10. studytime - weekly study time (numeric: 1 - <2 hours, 2 - 2 to 5 hours, 3 - 5 to 10 hours, or 4 - >10 hours)
11. health - current health status (numeric: from 1 - very bad to 5 - very good)
12. failures - number of past class failures (numeric: from 0 to 3, for more choose 4)
13. absences - number of school absences (numeric: from 0 to 40)
14. G1 - first period grade (numeric: from 0 to 20)
15. G2 - second period grade (numeric: from 0 to 20)

The class is the third semester grade:

1. G3 - final grade (numeric: from 0 to 20, output target)

Below (Code Listing 4.1) is the code for reading the data and the feature vector and target variable, for the math student dataset. For the reading of the dataset files, I used the pandas library.



```
# Read and preview data
df = pd.read_csv('student-math.csv')

# Declare feature vector and target variable
X = df[
    ['school', 'sex', 'age', 'address', 'famsize', 'Pstatus', 'Medu',
'Fedu', 'traveltime', 'studytime', 'health',
     'failures', 'absences', 'G1', 'G2']]
y = df['G3']
```

Code Listing 4.1 Reading of data and feature vector and target variable declaration.

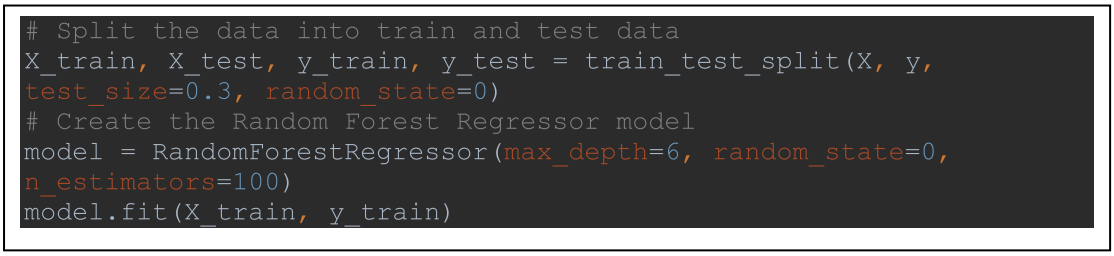
4.3 Classification Model

The Random Forest Regressor model, which is a black-box model, was selected to build a binary classification model for predicting the semester grade for the students.
Random Forest is an ensemble machine learning technique capable of performing both regression and classification tasks using multiple decision trees and a statistical technique called bagging. A Random Forest instead of just averaging the prediction of trees it uses two key concepts that give it the name random:

- Random sampling of training observations when building trees
- Random subsets of features for splitting nodes

In other words, Random Forest builds multiple decision trees and merge their predictions together to get a more accurate and stable prediction rather than relying on individual decision trees.[18]

Below (Code Listing 4.2) is the implementation of the Random Forest Regressor. We first split the data into train and test data and then create the model. The split was 70% training data and 30 % test data. For this, I imported train_test_split from the sklearn.model_selection library and RandomForestRegressor from the sklearn.ensemble library.

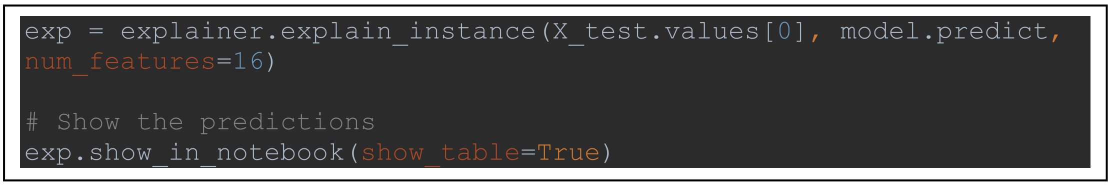


```
# Split the data into train and test data
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.3, random_state=0)
# Create the Random Forest Regressor model
model = RandomForestRegressor(max_depth=6, random_state=0,
n_estimators=100)
model.fit(X_train, y_train)
```

Code Listing 4.2 Random Forest Regressor implementation.

4.4 LIME

I used the LIME package which is written in Python and is available on GitHub[25]. To create the explanation and show it I used the following code.



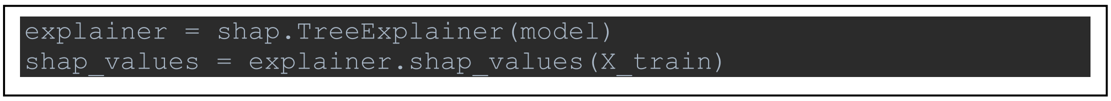
```
exp = explainer.explain_instance(X_test.values[0], model.predict,
num_features=16)

# Show the predictions
exp.show_in_notebook(show_table=True)
```

Code Listing 4.3 LIME prediction code.

4.5 SHAP

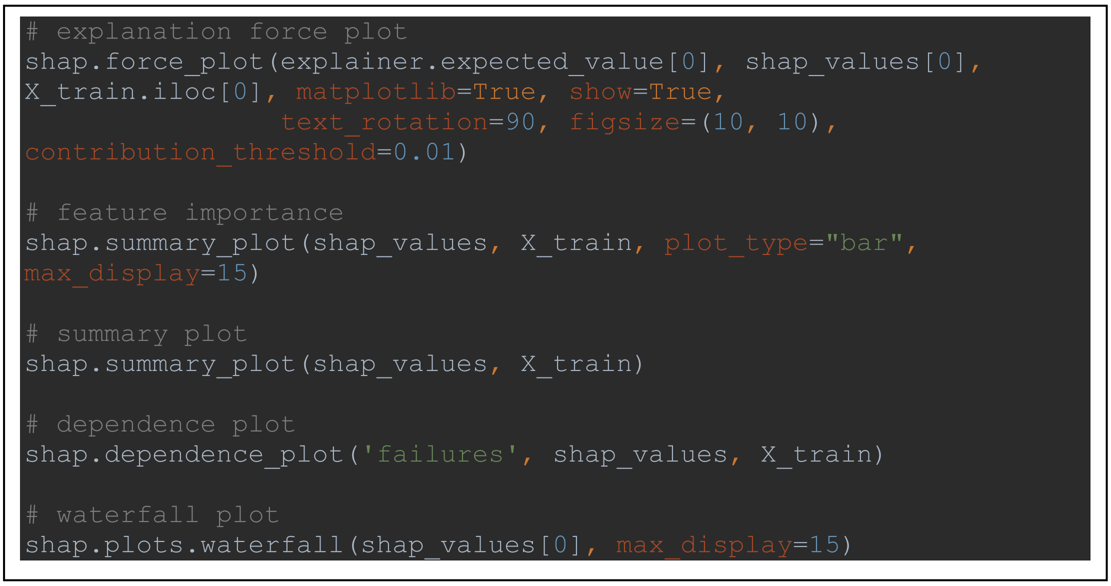
I used the SHAP package which is written in Python and is available on GitHub[17]. To create the explanation I used the following code.



```
explainer = shap.TreeExplainer(model)
shap_values = explainer.shap_values(X_train)
```

Code Listing 4.4 SHAP explanation code.

To create all the plots in Chapter 5 I used the methods below.



```
# explanation force plot
shap.force_plot(explainer.expected_value[0], shap_values[0],
X_train.iloc[0], matplotlib=True, show=True,
                text_rotation=90, figsize=(10, 10),
contribution_threshold=0.01)

# feature importance
shap.summary_plot(shap_values, X_train, plot_type="bar",
max_display=15)

# summary plot
shap.summary_plot(shap_values, X_train)

# dependence plot
shap.dependence_plot('failures', shap_values, X_train)

# waterfall plot
shap.plots.waterfall(shap_values[0], max_display=15)
```

Code Listing 4.5 SHAP plot methods.

Chapter 5

Grade Prediction Experiment

5.1 Grade Prediction

The experiment was set up to use two datasets of students to calculate their third semester grade using LIME and SHAP. One dataset was for students that took the Math course while the other was for the Portuguese course. Both sets of datasets have the same set of input features but for a different sample of students.

5.2 Importance and Dependence

Using SHAP, we begin by showing the Summary Plot (Figure 5.1 & Figure 5.2) which as we said in Chapter 4 this plot combines feature importance with feature effects. In both figures for the two courses, it shows that G2 is by far the most important feature that is taken into account when predicting the final semester grade, much more than G1 because of course a more recent grade is a better showcase of a student's current ability.

We can see that G2's high feature value has a high and positive impact on the final grade while also having a low and negative impact on it. Failures on the other hand have a high impact on those that have a lot of failures but very little impact on those that do not have failures.

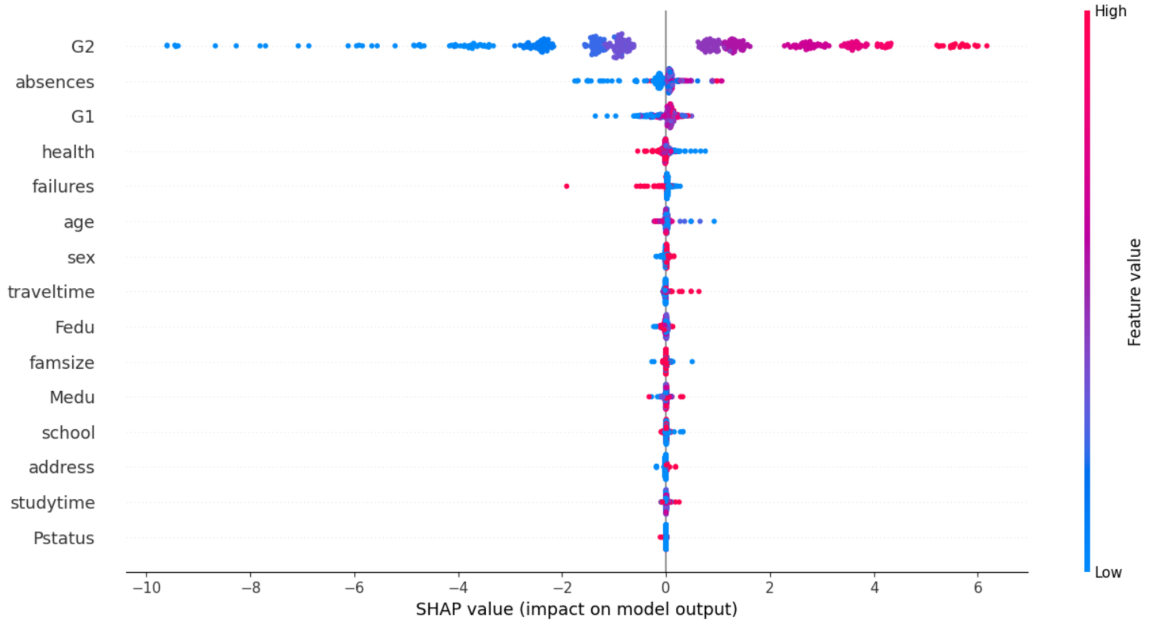


Figure 5.1 The Summary Plot for the Portuguese course

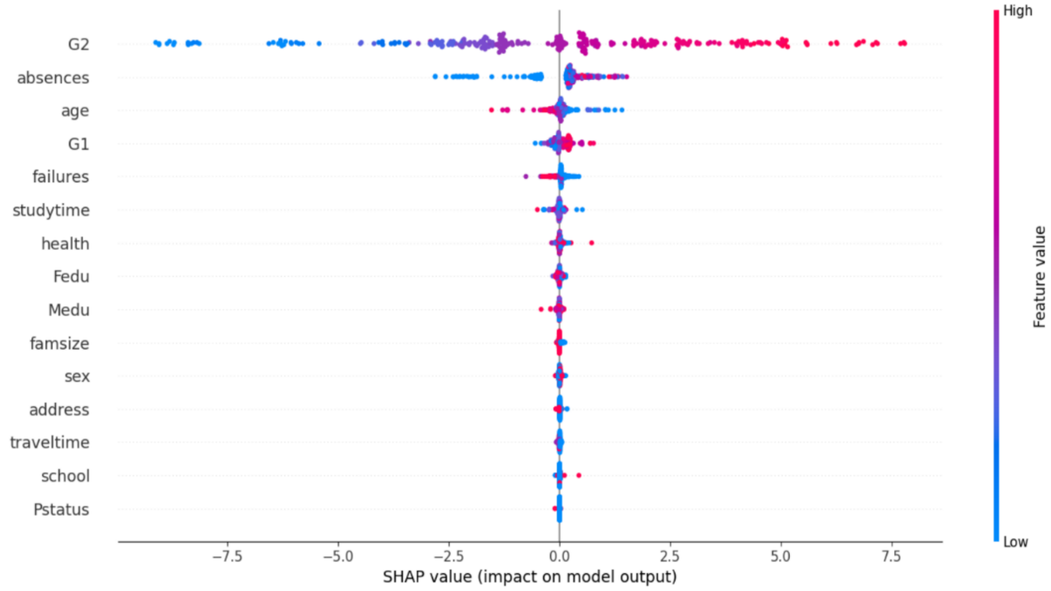


Figure 5.2 The Summary Plot for the Math course

Similarly, the feature importance plot (Figure 5.3 & Figure 5.4) shows the importance of each feature in a bar graph, where we can once again see how important G2 is to the final grade.

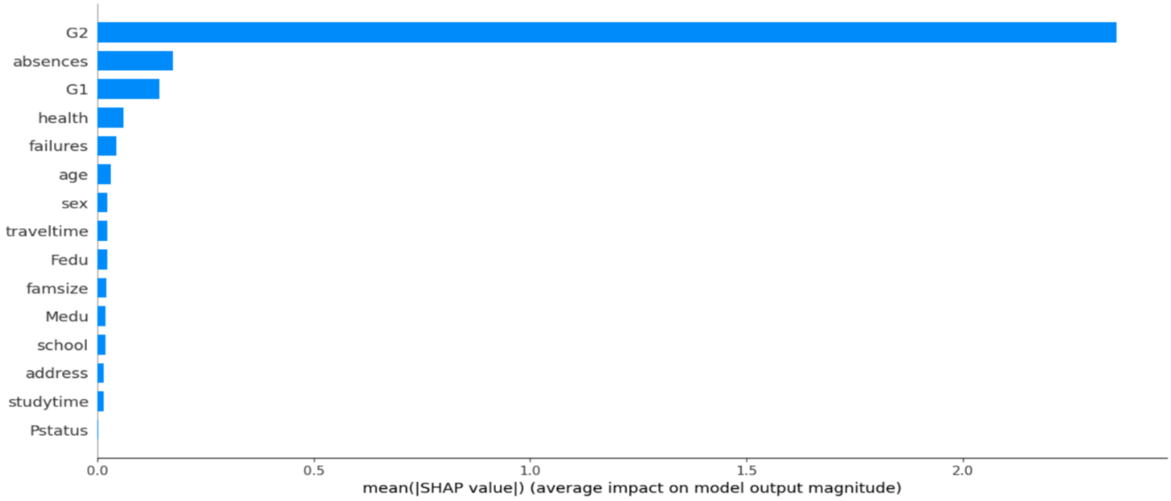


Figure 5.3 The Feature Importance Plot for the Portuguese course

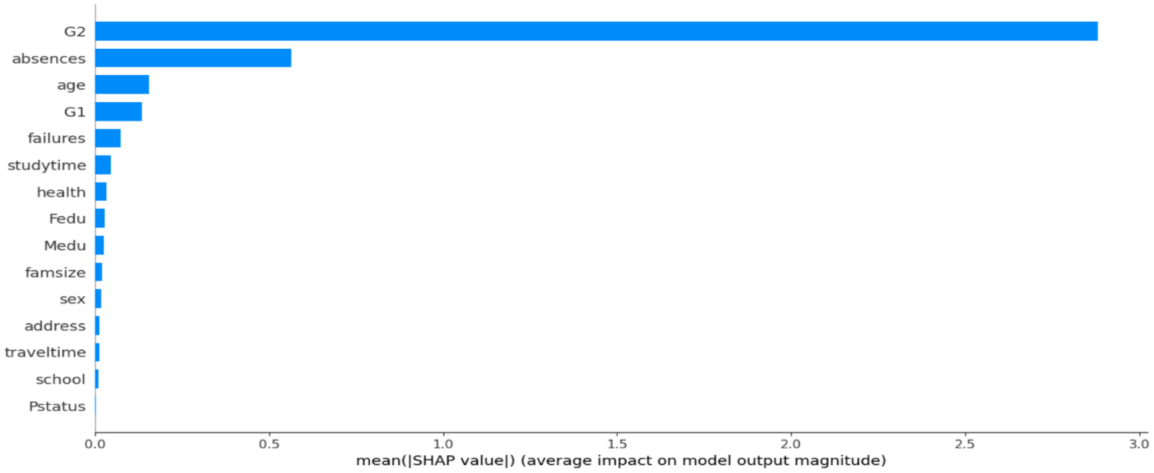


Figure 5.4 The Feature Importance Plot for the Math course

Dependence Plot in Figure 5.5 shows that G1 interacts the most with G2 and Figure 5.6 shows that failures interacts the most with G1.

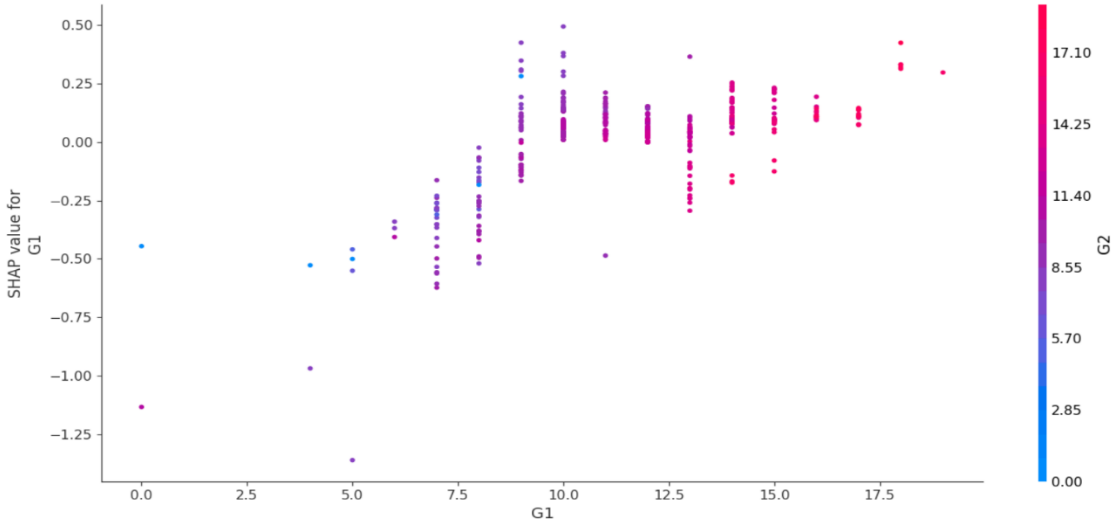


Figure 5.5 The Dependence Plot of G1 for the Portuguese course

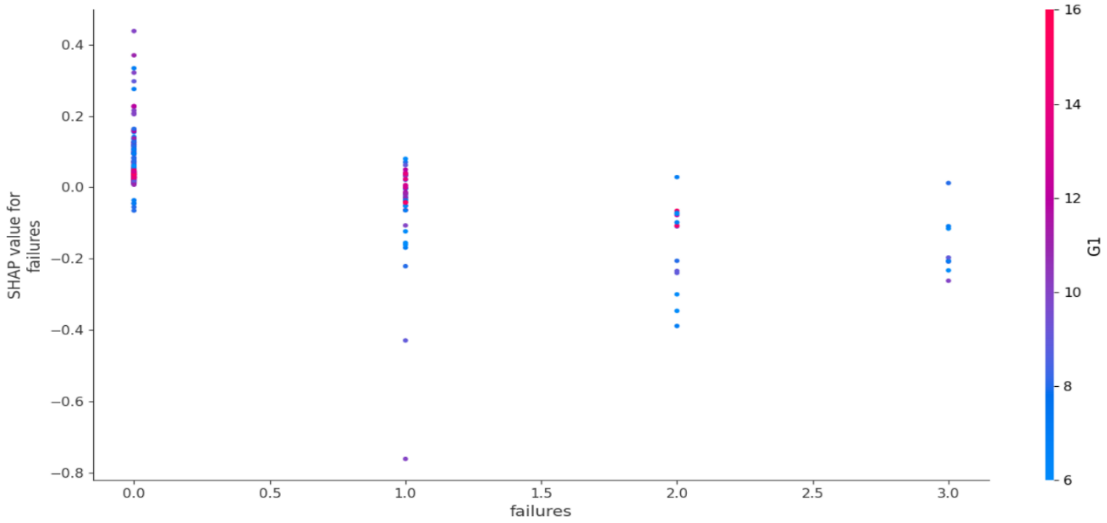


Figure 5.6 The Dependence Plot of failures for the Math course

5.3 Prediction Explanation

For these predictions I randomly inserted the below data in the two examples, in the first test data instance in order to create a prediction with my own input.

Input of Portuguese course student:

1. school = 0 (Gabriel Pereira)
2. sex = 1 (female)
3. age = 17
4. address = 0 (urban)
5. famsize = 0 (less or equal to 3 family members)
6. Pstatus = 1 (apart)
7. Medu = 2 (5th to 9th grade)
8. Fedu = 3 (secondary education)
9. traveltime = 1 (<15 min.)
10. studytime = 1 (<2 hours)
11. health = 3 (ok)
12. failures = 3
13. absences = 17
14. G1 = 12
15. G2 = 15

Our model was built on the portuguese course dataset. Then we apply LIME and the outcome is displayed in Figure 5.7 which shows the positive or negative impact that each feature has on the class variable, the semester grade (G3). G2 and absences have a positive impact while pstatus (the parent’s cohabitation status) and failures have a negative impact. The predicted value of the third semester for the particular student record is 15.49.

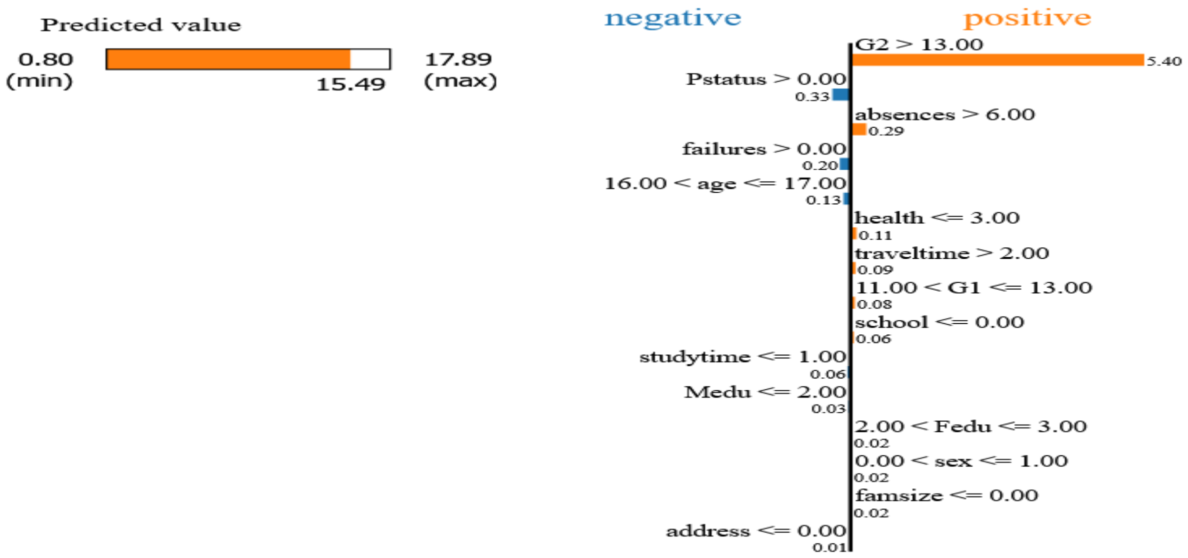


Figure 5.7 Explanation for a prediction of a portuguese course student

Looking at the SHAP Waterfall plot (Figure 5.8) we see the same reuslt where G2 has a positive and high impact, failures has a negative and low impact, and the predicted grade is 15.49.

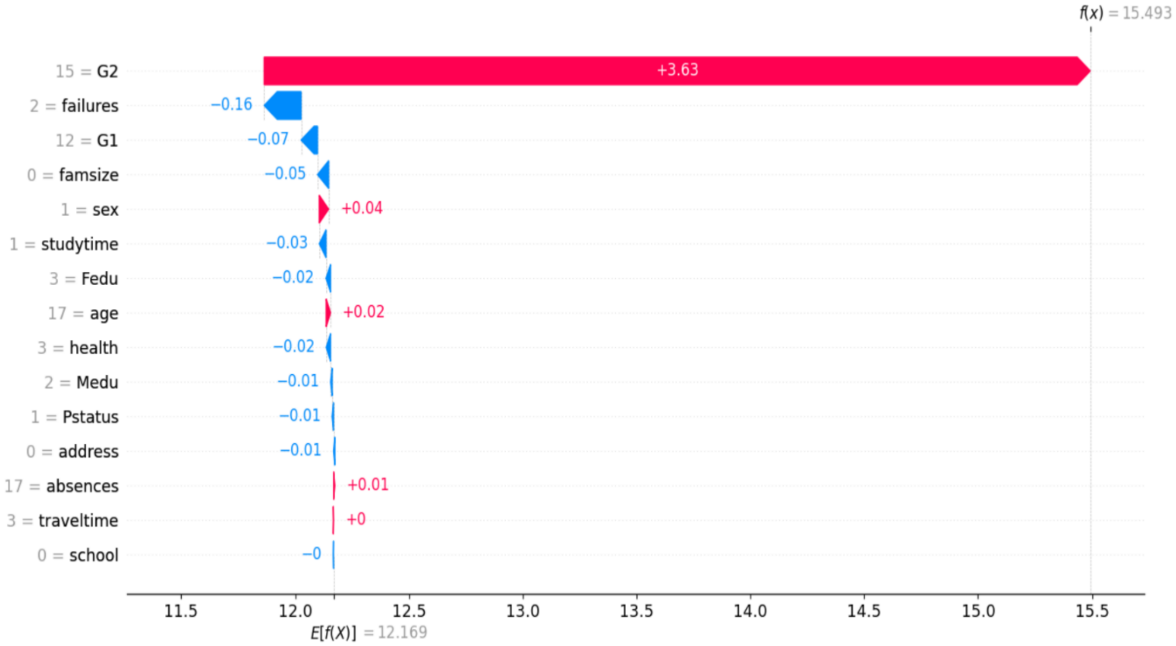


Figure 5.8 Waterfall Plot for a prediction of the semester grade for a student selected from the portuguese course dataset

Input of Math course student:

1. school = 1 (Mousinho da Silveira)
2. sex = 0 (male)
3. age = 20
4. address = 0 (urban)
5. famsize = 0 (less or equal to 3 family members)
6. Pstatus = 0 (living together)
7. Medu = 2 (5th to 9th grade)
8. Fedu = 2 (2 – 5th to 9th grade)
9. traveltime = 1 (<15 min)
10. studytime = 4 (>10 hours)
11. health = 2 (bad)
12. failures = 3
13. absences = 27
14. G1 = 17
15. G2 = 10

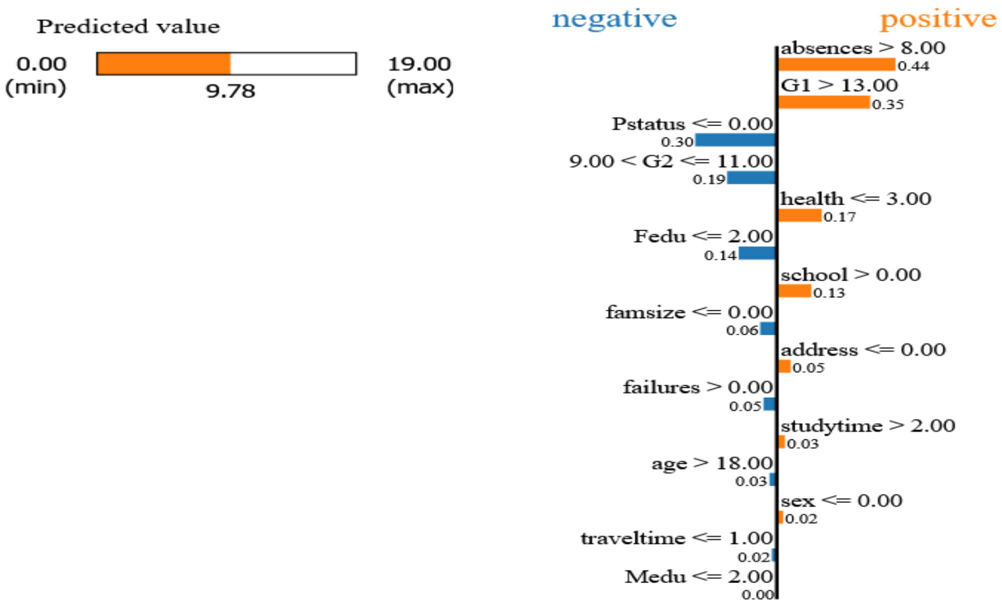


Figure 5.9 Explanation for a prediction of a math course student

is the second classification model was build on the math course dataset. Then LIME was applied to justify the predicted semester grade for a particular student selected from the dataset. LIME in Figure 5.9 shows us that G1 and absences have a positive impact while G2 and fedu (father's education) have a negative impact. We also see the predicted value of the third semester grade for this input data which is 9.78.

Looking at the SHAP Waterfall plot (Figure 5.10) we see the same result where absences has a positive and high impact, G2 has a negative and low impact and the predicted grade is 9.78.

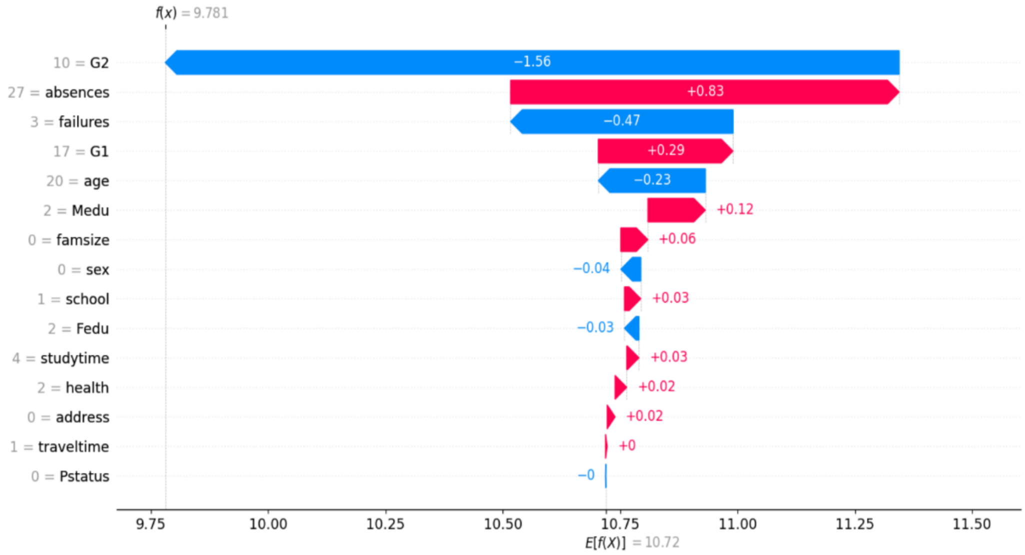


Figure 5.10 Waterfall Plot for a prediction of a math course student

Chapter 6

Conclusions

6.1 Experiment discussion

After applying Random Forest Regressor on both the datasets to predict the grade of the student, then we use two explainability techniques LIME and SHAP to understand the decision-making process for a particular student. we concluded that SHAP had the best interpretability of its explanation output in comparison to LIME, which did not provide as many meaningful explanations. LIME is great for single prediction explanation, but SHAP can give a single prediction explanation, together with a handful of plots that focus on the dependence and importance of features for the entire model.

So, if SHAP can do what LIME does but even better then why even bother using LIME? The answer is speed. LIME perturbs data around an individual prediction to build a model, while SHAP has to compute all permutations globally to get local accuracy.[16] For my implementation though, the dataset was not big enough, so LIME's advantage in speed made no difference in the end.

6.2 Summary

In this thesis, we talked about AI and its subsets, ML and DL. We learned what XAI is and the purpose it serves. We explained two model agnostic methods in LIME and SHAP through a different variety of examples and then we put those methods to use with our own chosen dataset, in order to predict a student's final semester grade.

6.3 What I have learned

The aim of the thesis was to learn about the explainability methods and put them to use. In Chapter 2 I gained knowledge about ML, DL and XAI. In Chapter 3 I saw examples of these explainability methods. And in Chapter 4 & 5 I used all I learned to run my own tests on a dataset I found interesting and discussed the results in Chapter 6.1.

6.4 Future Work

XAI is growing rapidly but still has some flaws. It is still in a state where we have to sacrifice accuracy for explainability. There are also other methods that we did not focus on here, like the What-if tool and ELI5. XAI’s future is one where we will create more explainable models and attempt at the same time to maintain a high performance. XAI will play a very important role in the coming decades, as it strides to get people to trust and understand the decisions made by AI systems.

Bibliography

[1] Mitchell, Tom M. ‘Does Machine Learning Really Work?’ AI Magazine 18, no. 3 (15 September 1997): 11–11. https://doi.org/10.1609/aimag.v18i3.1303.

[2] Jordan, M. I., and T. M. Mitchell. ‘Machine Learning: Trends, Perspectives, and Prospects’. Science 349, no. 6245 (17 July 2015): 255–60. https://doi.org/10.1126/science.aaa8415.

[3] Angelov, Plamen P., Eduardo A. Soares, Richard Jiang, Nicholas I. Arnold, and Peter M. Atkinson. ‘Explainable Artificial Intelligence: An Analytical Review’. WIREs Data Mining and Knowledge Discovery 11, no. 5 (2021): e1424. https://doi.org/10.1002/widm.1424.

[4] Xu, Feiyu, Hans Uszkoreit, Yangzhou Du, Wei Fan, Dongyan Zhao, and Jun Zhu. ‘Explainable AI: A Brief Survey on History, Research Areas, Approaches and Challenges’, 563–74, 2019. https://doi.org/10.1007/978-3-030-32236-6_51.

[5] Guidotti, Riccardo, Anna Monreale, Salvatore Ruggieri, Franco Turini, Dino Pedreschi, and Fosca Giannotti. ‘A Survey Of Methods For Explaining Black Box Models’. ArXiv:1802.01933 [Cs], 21 June 2018. http://arxiv.org/abs/1802.01933.

[6] Guidotti, Riccardo, Anna Monreale, Salvatore Ruggieri, Dino Pedreschi, Franco Turini, and Fosca Giannotti. ‘Local Rule-Based Explanations of Black Box Decision Systems’. ArXiv:1805.10820 [Cs], 28 May 2018. http://arxiv.org/abs/1805.10820.

[7] Sarker, Iqbal H. ‘Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions’. SN Computer Science 2, no. 6 (November 2021): 420. https://doi.org/10.1007/s42979-021-00815-1.

[8] P. Cortez and A. Silva. Using Data Mining to Predict Secondary School Student Performance. In A. Brito and J. Teixeira Eds., Proceedings of 5th FUture BUsiness TEChnology Conference (FUBUTEC 2008) pp. 5-12, Porto, Portugal, April, 2008, EUROSIS, ISBN 978-9077381-39-7.

[9] Kucak, D[anijel]; Juricic, V[edran] & Dambic, G[oran] (2018). Machine Learning in Education - a Survey of Current Research Trends, Proceedings of the 29th DAAAM International Symposium, pp.0406-0410, B. Katalinic (Ed.), Published by DAAAM, International, ISBN 978-3-902734-20-4, ISSN 1726-9679,Vienna,Austria DOI: 10.2507/29th.daaam.proceedings.05

[10] Ertel, Wolfgang. Introduction to Artificial Intelligence. Undergraduate Topics in Computer Science. Cham: Springer International Publishing, 2017. https://doi.org/10.1007/978-3-319-58487-4.

[11] Barredo Arrieta, Alejandro, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador Garcia, et al. 'Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI'. Information Fusion 58 (1 June 2020): 82–115. https://doi.org/10.1016/j.inffus.2019.12.012.

[12] 'Opening the Black Box: The Promise and Limitations of Explainable Machine Learning in Cardiology | Elsevier Enhanced Reader'. Accessed 2 May 2022. https://doi.org/10.1016/j.cjca.2021.09.004.

[13] Guestrin, Marco Tulio Ribeiro, Sameer Singh, Carlos. 'Local Interpretable Model-Agnostic Explanations (LIME): An Introduction'. O'Reilly Media, 12 August 2016. https://www.oreilly.com/content/introduction-to-local-interpretable-model-agnostic-explanations-lime/.

[14] Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. '"Why Should I Trust You?": Explaining the Predictions of Any Classifier'. ArXiv:1602.04938 [Cs, Stat], 9 August 2016. http://arxiv.org/abs/1602.04938.

[15] Burzykowski, Przemyslaw Biecek and Tomasz. 9 Local Interpretable Model Agnostic Explanations (LIME) | Explanatory ModelAnalysis.. https://ema.drwhy.ai/LIME.html#LIMEProsCons.

[16] Lundberg, Scott M, and Su-In Lee. ‘A Unified Approach to Interpreting Model Predictions’, n.d., 10.

[17] Molnar, Christoph. Interpretable Machine Learning. Accessed 3 May 2022. https://christophm.github.io/interpretable-ml-book/.

[18] Segal, Mark R. ‘Machine Learning Benchmarks and Random Forest Regression’, 14 April 2004. https://escholarship.org/uc/item/35x3v9t4.

[19] Abdul Wahid. ‘Big Data and Machine Learning for Businesses’. 03:42:34 UTC. https://www.slideshare.net/awahid/big-data-and-machine-learning-for-businesses.

[20] David Gunning. Explainable Artificial Intelligence (XAI). Technical report, 2017. URL https://www.darpa.mil/attachments/XAIProgramUpdate.pdf.

[21] Samuel AL. Some studies in machine learning using the game of checkers. IBM: J Res Dev. 1959;3:210–29.

[22] Li W., Chai Y., Khan F., et al. A comprehensive survey on machine learning-based big data analytics for IoT-enabled smart healthcare system. Mobile Networks and Applications. 2021;26(1):234–252. doi: 10.1007/s11036-020-01700-6.

[23] R V, Belfin, E. Kanaga, and Suman Kundu. ‘Application of Machine Learning in the Social Network’, 61–83, 2020. https://doi.org/10.1002/9781119551621.ch4.

[24] Adadi, Amina, and Mohammed Berrada. ‘Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)’. IEEE Access 6 (2018): 52138–60. https://doi.org/10.1109/ACCESS.2018.2870052.

[25] Ribeiro, Marco Tulio Correia. *Lime*. JavaScript, 2022. https://github.com/marcotcr/lime.

[26] ‘Home Page for 20 Newsgroups Data Set’. Accessed 11 May 2022. http://qwone.com/~jason/20Newsgroups/.

[27] 'UCI Machine Learning Repository: Mushroom Data Set'. Accessed 11 May 2022. https://archive.ics.uci.edu/ml/datasets/mushroom.

[28] 'Google Colaboratory'. Accessed 11 May 2022. https://colab.research.google.com/github/arteagac/arteagac.github.io/blob/master/blog/lime_image.ipynb.

[29] 'Boston Dataset'. Accessed 11 May 2022. https://www.cs.toronto.edu/~delve/data/boston/bostonDetail.html.

[30] Wolf, Thomas, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Perric Cistac, et al. *Transformers: State-of-the-Art Natural Language Processing*. Python. 2018. Reprint, Association for Computational Linguistics, 2020. https://www.aclweb.org/anthology/2020.emnlp-demos.6.

[31] 'UCI Machine Learning Repository: Student Performance Data Set'. Accessed 11 May 2022. https://archive.ics.uci.edu/ml/datasets/Student+Performance#.