

Natural Language Explanations

Author
David-Paul Niland

Primary supervisor
Albert Gatt

Secondary supervisor
Pablo Mosteiro Romero



**Utrecht
University**

MSc Artificial Intelligence
Graduate School of Natural Sciences
Utrecht University
Netherlands
August 2022

Dedication

To Mum, Dad and Mary

Declaration

I declare that this thesis has been composed solely by myself and that it has not been submitted, in any previous application for a previous degree or qualification. Except where states otherwise by reference or acknowledgement, the work presented is entirely my own.

Acknowledgements

I want to thank all the continued help and support I received during my master's thesis research. Most importantly, I would like to thank my primary supervisor Albert Gatt, who has been a wealth of information. His support and enthusiasm have been ever helpful, even during the global pandemic. I also thank my two external supervisors at Avaya, Gary Collins and Aaron Hurley. I am forever grateful to them both. They are fountains of knowledge from which I have learned plenty. I would also like to thank my honorary supervisor, Ettore Mariotte. I wish you the very best of luck in the rest of your PhD and research.

Abstract

This thesis is focused on generating natural language explanations for Automated machine learning (AutoML). Research in natural language explanations is timely, given both the popularity of explainability techniques and the continued advances in AutoML. We believe that the standard explainability techniques are not explicit enough in conveying information to stakeholders. Users might prefer one mode of information over another [43] or feel more confident with visual information [13]. In other domains, people understand information better if it is presented with it in natural language [13], [43]. We have therefore proposed, developed and tested language generation modules that build explanations for machine learning models that can be applied to AutoML systems. This research provides a bedrock for future work on generating natural language explanations.

We have developed three language generation modules for permutation feature importance, partial dependence and accumulated local effects. During the development of the language generator modules, we conducted a preliminary pilot study to evaluate the systems. This study helped the development process and deepened our understanding of the language required to explain the graphical information. To test whether natural language explanations can offer more utility than visual explanations, we conducted a more extensive evaluation study to test which mode of explanation was more helpful: visual, textual or multimodal. What constitutes a "good" explanation is one that helps users understand the underlying information that is being conveyed. In this thesis, study participants found multimodal explanations to be the most useful of the three modes in increasing their understanding of the underlying processes.

Table of Contents

Chapter 1	7
Introduction	7
1.1. Research question and thesis goals	7
Chapter 2	9
Literature Review	9
2.1. AutoML	9
2.2. Explainable AI	10
2.2.0.1. To justify	10
2.2.0.2. To control	10
2.2.0.3. To improve	11
2.2.0.4. To discover	11
2.2.1. Permutation feature importance	11
2.2.2. Partial dependence plots	12
2.2.3. Accumulated local effects plots	12
2.3. Explainability in autoML	13
2.4. Issues with explainability	13
2.5. Natural language generation	14
2.6. Natural language explanations	15
2.7. Multimodal explanations	15
2.8. Evaluation	16
2.8.1. Evaluation checklist	16
Chapter 3	18
Methods	18
3.1. Implementation	18
3.2. Data	18
3.2.1. Cervical cancer	18
3.2.2. Australian weather data	19
3.2.3. National basketball association	19
3.3. Classification Models	19
3.4. Natural language generation module	20
3.4.1. Permutation Feature Importance	20
3.4.1.1. Template for permutation feature importance	20
3.4.2. Partial Dependence and Accumulated Local Effects	22
3.4.2.1. Template for partial dependence and accumulated local effects	22
3.4.3. Local Regression	25
3.5. Evaluation	26
3.5.1. Preliminary pilot study	26
3.5.1.1. Participants	28
3.5.1.2. Design	28
3.6. Results of preliminary pilot study	29
3.6.1. Quality	29
3.6.2. Suggestions	31
3.6.2.1. Permutation Feature Importance	31
3.6.2.2. Partial Dependence Plots	31
3.6.2.3. Accumulated Local Effects Plots	31
3.6.2.4. Summary of results	31

3.7.	Implementation Part 2	32
3.7.1.	Permutation feature importance	33
3.7.1.1.	New template for permutation feature importance	33
3.7.1.2.	Clustering	33
3.7.1.3.	Changes to lexicalisation	34
3.7.2.	Partial dependence and accumulated local effects	34
3.7.2.1.	Templates for partial dependence and accumulated local effects	34
Chapter 4		39
Results		39
4.1.	Evaluation	39
4.1.1.	Main study	39
4.1.2.	Participants	39
4.2.	Procedure	39
4.3.	Design	39
4.3.1.	Materials	39
4.4.	Results	40
4.4.1.	Removal of outliers based on the log-scale	44
4.4.2.	Results	44
4.4.3.	ANCOVA	44
Chapter 5		48
Discussion		48
5.0.1.	Local methods	48
5.0.1.1.	Clustering	49
Chapter 6		50
Conclusion		50
Appendix A		54
Appendix		54
A.1.	Main study	54
A.1.1.	Permutation feature importance	54
A.1.1.1.	Cervical cancer dataset	54
A.1.1.2.	NBA dataset	54
A.1.1.3.	Rain dataset	54
A.1.2.	Partial dependence	54
A.1.2.1.	Cervical cancer dataset	54
A.1.2.2.	NBA dataset	54
A.1.2.3.	Rain dataset	54
A.1.3.	Accumulated local effects	54
A.1.3.1.	Cervical cancer dataset	54
A.1.3.2.	NBA dataset	54
A.1.3.3.	Rain dataset	54

Chapter 1

Introduction

Automated Machine Learning (AutoML) is the process of automating the machine learning model development to alleviate the user of the tedious iterative process that relies heavily on human expertise. The adoption of AutoML will give greater power to a larger demographic and make machine learning more accessible to a broader audience instead of solely for engineers and researchers. Although the transition will provide a net good to the research area, it is not without issues, particularly when applying black-box algorithms to high-risk situations. For this reason, greater emphasis should be put on the end-user to help understand the inner workings of the AutoML process.

Despite all the progress it has seen in recent years, there is a lack of transparency and interpretability in machine learning. There is a need to increase trust in machine learning models' decisions in healthcare, finance, and law [8]. Although the terms *explainability* and *interpretability* are sometimes confused, a clarification is necessary to avoid confusion. Interpretable machine learning refers to the methods and models that make the behaviour of machine learning systems understandable to humans. Although some models are intrinsically interpretable, (such as logistic regression or generalised additive models) others are not (neural networks, random forests). Explainability techniques are a set of techniques that can be applied to machine learning models to make their processes more interpretable and more similar to intrinsically interpretable models.

This project is a collaborative effort between Avaya and Utrecht University. Avaya is a multinational business communications software and cloud solutions company that has developed a working AutoML system. Some decisions undertaken are because of the specifications of the AutoML system, namely the decision to focus on global model-agnostic methods over local model agnostic methods. These methods will be elaborated on further under the explainability section of this paper 2.2.. The system is an application with a clean user interface and allows the user to drag and drop a dataset into the system. The user then picks the target variable. This step is followed by the system producing a set of candidate feature variables that it deems to influence the target variable. Following the feature confirmation, the system finds a candidate model from the search space, trains the model and returns accuracy, precision, recall and the F1 measure. Before training and evaluation, the user selects variables from a set of auto-generated candidates and decides which model to select. AutoML not only automates the process and lowers the barrier to entry into machine learning but often finds more accurate models than the traditional machine learning approach [16]. Explainability techniques can be a way to provide feedback on the processes within the AutoML system [2] and can allow them to trust the models that have been selected, as well as the ability to gain more insight into the feature variables that the users have chosen [29, 22]. It could be the case that the model heavily relies on an undesirable feature variable such as race, gender or religion, which could have extremely negative consequences in certain circumstances.

1.1. Research question and thesis goals

This research will ask whether natural language is more informative at providing explanations than graphical methods in the context of AutoML? We hypothesise that users can be better informed by textual information than graphs, which is in line with other research in natural language generation [Law, 40], [13]. As a secondary research question, this project will examine if a multimodal approach to explainability will increase participants' understanding of the underlying processes in AutoML. It might be the case that having both the visual graph explanation and generated text will help the end-user better understand the complex processes of the underlying algorithms [30]. In doing so, there will be three conditions for the experiment, where unimodal explanations are evaluated individually and a combination of both conditions to see if visual graphs combined with generated text give AutoML users a better understanding of the process overall. We hope this research will not only apply to the application space of AutoML but to machine learning more broadly.

The research questions of this project, therefore, are the following:

This work will address the following questions in terms of user understanding of explanations in the AutoML context:

- Are textual explanations more effective than visual explanations?
- Are visual and textual explanations combined more effective than individual modes?

This thesis will provide an overview of the field of AutoML and Explainability. We selected a subset of explainability methods and discussed why we think global model agnostic techniques are most applicable to the project. Then, we will describe how some explainability techniques are flawed, particularly when considering AutoML users. The methods section will cover how we addressed the research question with our language generation modules and tested our research questions. This research can be used as an exploratory roadmap for other work in natural language explanations, which is a sparsely researched field at the time of writing.

Chapter 2

Literature Review

2.1. AutoML

In the standard approach to machine learning, a researcher has to perform many tasks that can be rather time-consuming and relies heavily on human expertise. AutoML is an attempt to automate this process and allow people to build high-quality machine learning systems. It can allow people with little to no prior experience in coding or machine learning to perform machine learning tasks without writing any code themselves. The resulting accuracies are often higher than what humans can obtain when developing the models themselves [16]. Many different tools and platforms are available today to help users [39]. Advances in computing power and better algorithms have led to more development in autoML among researchers and industry alike. The automatic preparing, cleaning data, feature engineering, discovering models, hyperparameter optimisation and evaluating the model are all steps that autoML researchers try to automate [42, 39].

Domain expertise is a crucial factor in the data science pipeline. Often data scientists do not have high-level knowledge for the fields which they are developing machine learning models for. This limitation is significant as domain expertise is an integral part of machine learning development. AutoML allows for complex algorithms to be put into the hands of these domain experts. Even for data scientists and machine learning engineers, autoML shows promise by saving them significant amounts of time while they concentrate on other tasks [12]. Algorithm selection and hyperparameter tuning can be laborious and take time to change architectures and settings iteratively. Engineers are only as skilled as the algorithms in their toolbox, and autoML means that the search space for which model, or combination of models, to select is much larger. Furthermore, data scientists can be biased towards using algorithms they are familiar with over algorithms that might better suit the data. AutoML can reduce this bias.

Many of the autoML systems that have been proposed are unfinished propositions of a completely automated autoML pipeline [16]. They are incomplete as they only automate certain blocks of the autoML process and others require some level of input from the user. In contrast to this, some provide end-to-end automated pipelines. VEGA is an example of an end-to-end autoML pipeline [41]. Although this might be welcome for some, perhaps an approach to autoML where the human is kept in the loop has an added benefit of allowing some decisions to be overseen by humans. Complete pipeline automation could mean more accurate systems, but perhaps full automation ignores domain expertise and undervalues human oversight. The entire workflow of the machine learning pipeline automatised might lead to a lack of control or understanding of the systems.

One of the main issues with autoML, as it stands today, is that it is costly to run algorithms and develop these systems at scale [41]. AutoML systems can be very taxing in terms of computing time. However, this will likely become less of an issue as hardware progresses. Nevertheless, autoML will likely become more of a mainstream tool for both machine learning engineers and laypeople in the future. There is a solid argument that part of the success of neural networks is due to their automated feature engineering, and in this way, when more of the process is automated, the barrier to entry is lowered.

Although autoML's goal is to take the human out of the loop, this can be seen as a major disadvantage of autoML [42]. Xanthopoulos et al. suggest that more of the process should focus on the human user. The end-users should be aware of and explain the inner workings of the autoML process. Doing so will, in turn, determine the success or failure of a system's wider adoption. The user should understand why a particular algorithm is better suited to one situation over another. For example, perhaps one algorithm uses the variable postcode as the most important variable in predicting whether a loan is given or not, which could be problematic.

There are many techniques in autoML systems for automatic model selection. Although there is no unifying approach for creating autoML systems [12], there has been significant research in recent years. This is largely due to advances in Combined Architecture Search and Hyperparameter optimisation (CASH), Neural Architecture Search [45, 16] and Evolutionary Algorithms [32]. In addition to this, work has been done to create more open-source benchmarks for autoML systems, which further increase public knowledge in the area and develop the field even further [12].

The autoML system on which this project is based focuses explicitly on supervised classification problems. The user loads a dataset into the system and selects the target variable, then, the system automatically selects what it finds to be the feature variables of interest. The user then confirms the feature selection before the machine learning algorithm is selected and the prediction made. However, if users do not understand the inner workings of how these predictions are made, then there is more potential for harm once the system is deployed. Therefore the models that these systems produce must be explained well.

2.2. Explainable AI

Explainable AI is a research field that aims to make AI systems results more understandable to humans [1]. Sometimes the terms interpretable AI and explainable AI are used interchangeably. In order to avoid confusion, a necessary clarification needs to be addressed. Interpretable machine learning is the field that encompasses explainable AI as a subfield. Interpretable models are so due to their inherent design. They are intrinsically interpretable by looking at their model parameters or feature summary statistics. Models that are not intrinsically interpretable are black box models that require external models to add insight post-hoc into their inner workings. Interpretable models can often be more desirable when interpretations are more important than accuracy [4]. Explainable AI comes from the need to justify machine learning algorithms that are not inherently interpretable. Black-box models are machine learning models that cannot be understood by looking at their parameters alone. Neural networks and random forests are typical examples of black-box models. The issue with black-box models is that they lack an explicit declarative knowledge representation, which means that they lack any underlying explanatory structures [17]. In opposition to this are interpretable models, which are sometimes called white-box models.

The utility of black-box models has come into question, and some argue that they should not be used for high stakes decisions at all [37]. Although there are some clear advantages to using inherently interpretable models, black-box models are still widely used, partly due to their ease of use and high accuracy. Although some dispute whether black-box models are always more accurate. The performance-interpretability trade-off is what this dispute has been coined. Research has been done into demystifying the performance-interpretability trade-off by using information from black boxes to inform interpretable models [15], which then perform with high accuracy while giving accurate interpretability. Although this work is a step in the right direction, whether interpretable models can consistently achieve higher accuracies than black boxes is inconclusive as of yet. However, their ease of use and high accuracy are compelling reasons to continue researching their value, given the ongoing debate. A better approach might be to see how explaining black boxes can be improved.

The need to explain or interpret comes from building trust in algorithms, particularly helpful when there are high stakes decisions and to build systems that coincide with our laws and values [9]. Algorithmic decisions and any data driving those decisions should be explained easily and effectively to end-users and other stakeholders. The need for explainable AI comes from many different areas. Adadi et al. highlight four main categories when considering the driving forces. The categories are the need to justify, control, improve and to discover [1].

2.2.0.1. To justify

There have been multiple controversies in recent years when machine learning or AI systems have had biased or discriminatory results. Explanations offer a way to ensure that the decisions made by algorithms were not made erroneously and that the decisions were fair and ethical. Explainable AI is a way to defend a decision of an algorithm so that errors are minimised, and trust in the algorithm is built. In addition, justifications can provide a way to comply with legislation. The right to explanation is part of the EU's General Data Protection Regulation act [14].

2.2.0.2. To control

Explainable AI can help from algorithmic decisions going wrong. It can do so by providing the user with a greater understanding of the system's unknowns and insight into errors, flaws, and vulnerabilities.

2.2.0.3. *To improve*

Explainable AI can be a way to improve models continuously. A model that is well explained can be improved as the users are better informed about it. This improvement can occur through an iterative process, and explainability techniques offer feedback for ongoing development.

2.2.0.4. *To discover*

Explanations can be a way to offer new facts and knowledge about what is being explained. In a recent study, Liu et al. found that their explanations led to the discovery of new facts [22]. Their research used an autoML system that used partial dependence, accumulated local effects, permutation feature importance and feature interaction post-hoc on the system. Using these explainability techniques, it allowed them to drill down into the data further than what the models were predicting by themselves. The techniques allowed them to find what factors contributed most to blood levels in childhood. They discovered that children had higher blood Pb levels if they lived within 1km of the central mining area or 1.37km to the railroad. As well as this, they discovered that year of testing was the feature variable that interacted with most other features, and blood Pb levels increased faster in Aboriginal than in non-Aboriginal children.

There have been some high profile cases of AI systems that have gone wrong in the past. One particular case was that of COMPAS, where AI was applied to the criminal justice system [23]. The algorithm gave a recidivism-risk score to arrested people. The algorithm's fairness was brought into question as the recidivism score was overestimated for black people. This case was a prime example of a case that required inquiry into the algorithm's decisions. Domains that explainable AI shows the most prominence for are medical [17], transportation [1] and legal [5].

There are different dimensions of Explainable AI. There are *model specific* methods and *model agnostic* methods. An example of a model-specific explainability technique would be pixel saliency maps in a convolutional neural network. Model agnostic methods can be applied to any machine learning model. There are global methods that describe the overall behaviour of the model on average and local methods describe individual predictions [25]. Although focusing this project on either local or global methods would be beneficial for the autoML system, global methods might be more beneficial to the system that this project is based on. They are crucial as they can provide feedback [2] to the user on the specific variables they have confirmed relevant. The global methods are particularly useful when the modeller wants to understand the general mechanisms in the data or debug a model. It is for this reason that they are more beneficial to autoML as the practitioner is likely comparing multiple models and how the models use the same data in different ways. The explainability techniques undergo the same evaluation process. This feature makes it easier to compare techniques across different models. Model agnostic methods give more flexibility in terms of model, explanation and representation [36], [1]. Model flexibility allows for the explainability technique to be applied to autoML. Explanation flexibility allows for different forms of explanation, graphic explanations, linear formulas, or in this project's case, natural language. Global methods are a good way to provide feedback to users about the feature variables that they have to select before running the autoML system [2]. Reducing the number of variables might lead to more accurate and robust models.

The methods chosen for the project are well-grounded, widely used and applicable to autoML [22, 29]. They are global, model-agnostic methods that are applied post-hoc after training. Having multiple model-agnostic explainability techniques might improve the understanding of the model behaviour overall. They can work in unison to help a better understanding of the entire process, rather than just one technique [1]. Selecting to experiment with multiple techniques is preferred as no one technique will apply to every situation [27]. The techniques considered are a) permutation feature importance, b) partial dependence plots, and c) accumulated local effects plots. These techniques work well with one another, and some might be very informative in situations where others are not [22].

2.2.1. *Permutation feature importance*

Permutation Feature Importance (PFI) measures the model's prediction error increase after the feature values are permuted. PFI breaks the relationship between the feature and the outcome. After the model has been fitted, if a single feature vector is chosen and shuffled randomly while

leaving all other columns in place, PFI then measures how much the model's accuracy will be affected [25]. The resulting explanation is in tabular format, which gives the range of increase with the random shuffling. The variables are seen as important if the shuffle leads to a significant decrease in accuracy. If the model accuracy does not change or only changes slightly, the feature is not important.

Although there are feature importance scores for models such as Random Forests, these will not be considered, as taking a model-specific approach could impinge on the applicability to autoML. One of the critical reasons permutation feature importance is suitable for explainability is that it provides insight into the model's global behaviour in a very concise format. It is also rapid as it does not require retraining of the model. Permutation importance considers both the feature importance and all interactions with other features.

Permuted feature importance is not the only metric we might consider when explaining a model. One of the main disadvantages is that it is linked to the model's error by design. Perhaps the model's output variance might be more beneficial to measure the robustness. If features are correlated, there could be bias in the permuted feature importance by introducing unlikely instances, similar to partial dependence plots.

2.2.2. *Partial dependence plots*

Partial Dependence Plots (PDPs) are one of the more widely used model-agnostic explainability techniques. They are pretty easy to implement and considered by some to be easy to understand [25]. Partial dependence plots help understand the marginal effect on the predicted outcome. They illustrate how the prediction changes as the value of the interested feature changes while considering all other features in the model. PDPs are low-dimensional graphical renderings that help users understand the relationship between the target and the features of interest.

Partial dependence plots make a large assumption that is not always applicable to every set of features. They assume that features are not correlated, which is not always realistic as features are often correlated. PDPs average over their predictions can often lead to artificial data instances that are unlikely in reality. An example of this might be in a dataset with the variables age and salary, where salary is averaged to give babies an average salary of €30,000 a year.

One paper found PDPs to be not nearly as informative as accumulated local effects plots as many of the features of interest were correlated [22]. Rather than ruling out PDP plots or ALE plots, both were taken into account for further experimentation throughout this project.

2.2.3. *Accumulated local effects plots*

Accumulated Local Effects Plots (ALE plots) describe how a feature affects the prediction on average [25]. They are a very close relative of partial dependence plots. Because. Unlike partial dependence's averaging step, ALE alleviate this as they look at the differences between predictions instead of averages, which is what PDPs do.

Another straightforward advantage ALE plots have over PDP plots is that they are less biased. They work well when features are correlated, which is often the case in machine learning. For this reason, they are often preferred over PDP plots. They also have a faster compute time relative to partial dependence plots. If two features do not interact, the plot shows nothing.

Although they have many advantages, they are not perfect and do not work in every situation. If features have a strong correlation coefficient, then the interpretation of the effects across intervals is not possible. When there are many intervals, the plots can become unstable. There is no perfect solution for setting the number of intervals in ALE plots. A smaller number of intervals will lead to an inaccurate ALE plot, whereas a high number of intervals will lead to a shaky curve.

Some research has shown that ALE plots can help users improve the accuracy of autoML by 7-8% by providing feedback while developing in the pipeline [2]. Their research targeted people who had little to no experience in machine learning. The study suggested that ALE plots can be a tool to understand better how users can improve their inputs to the system by leveraging their domain expertise. In this way, ALE plots are incredibly useful because often, there is no path to improve the autoML process for laypeople.

2.3. Explainability in autoML

Although some of the more well-known autoML systems have explainability techniques as standard [39], not all do. There is limited research on the crossover between explainable AI and autoML; only a few papers highlight the area. Two in particular use autoML systems that provide model-agnostic explainability techniques in order to explain more about the features leading to a prediction [29], [22]. Both papers have a strong emphasis on domain expertise. One is specific to precision fish farming, the other sheds insight into features contributing to childhood blood lead levels.

Xanthopoulos et al. [42] regard interpretability as the most crucial factor in selecting an autoML service. They perform a qualitative study, which compares some of the autoML services. What they include in their analysis as constituting interpretability includes more than what interpretability is defined as elsewhere [25]. However, their conclusion remains valid; there is not nearly enough interpretability available to autoML users. Perhaps interpretability in the autoML sense should take a complete approach where data visualisation, progress report, and feature selection are considered under the umbrella of interpretability.

Their study's final feature set interpretation section is the most connected area to this project. Xanthopoulos et al. view the final feature set interpretation as helping the user select feature functionality [42]. In their final feature set interpretation mechanisms, they include: a) random forest feature importance ranking, b) LOCO feature importance, c) partial dependence plots, d) SHAP plots, e) ICE plots, f) a report of the standardised individual and cumulative importance of the participating features, g) the standardised coefficient for each feature in the case of a linear model and h) information about the resulted feature sets in the case of multiple feature selection. ICE plots are similar to PDP, only instead of showing averages, they show individual lines for individual instances prediction. Leave-One-Covariate-Out (LOCO) feature importance follows the same objective as permutation feature importance, although instead of permuting the feature values, it leaves the feature out entirely. Of the autoML systems that Xanthopoulos et al. reviewed, only two out of the seven studied included four or more techniques [42]. Only two received a grade of B, which was given if the autoML service had more than two final feature set interpretations. The final three autoML systems were awarded a grade of C, which meant that at least one technique was included. Including only one or two explainability techniques does not inspire confidence that the model will be explained sufficiently. Even though the goal of autoML is to take the human expert out of the loop, one could view this as a disadvantage. In autoML research there is perhaps too much focus on predictive performance and this strategy ignores the user experience. In their survey of autoML, He et al. [16] discuss how although there have been significant advances in how configuration settings can be found more efficiently for machine learning algorithms than humans, they highlight how there is a lack of understanding of why this is the case.

It seems that there is very little to allow autoML users to understand what happens under the hood in an autoML system. There is not enough on allowing autoML to be adopted by people who might not understand much about machine learning in general.

2.4. Issues with explainability

What constitutes a "good" explanation? Explanations should be audience-friendly, faithful to the system's decision process, and better interact with users. In this way, explanations could help improve AI and human decision-making. Kaur et al. highlight that in their experimental research, few of their data scientist participants were able to accurately describe the output of the feature visualisation tools accurately [19]. The study found that data scientists often over-trust and misuse interpretability methods (GAMs) and explainability methods like SHAP. It begs whether the conventional methods for feature explanations are sufficient or whether better methods need to be employed. Although Kaur's work is on glass-box models (GAM) and black-box model local model agnostic techniques (SHAP), we feel that this paper is a good indication of a lack of understanding of both interpretability and explainability techniques more broadly. GAMs are statistically-based models that have some similarities to PDPs in their plots. However, GAMs are much more trustworthy than PDPs as they are intrinsic. They do provide both global explanations and local explanations. SHAP are Shapley Additive explanations that exemplify a local model-agnostic technique. The participants in the experiment were even given standard tutorials on the visualisations and could not

explain manipulated nonsensical explanations. This finding begs whether the interpretability techniques are doing a good job explaining the models. Their research called for a more user-centric approach to evaluating interpretability tools in machine learning. To counter this confusion and apparent lack of understanding, it seems like a further emphasis on HCI needs to be employed to explain machine learning algorithms to people better. The best way of doing this is to perform a user study to see if an alternate method of explaining can better inform users. The Kaur et al. study used data scientists for their participants, who received a tutorial on the interpretability techniques and still did not understand how to interpret them correctly. Then what hope do users of autoML systems have to understand interpretability? Particularly when considering autoML users who may have minimal understanding of machine learning in general. Kaur et al. hypothesise that perhaps visualisations allow people to think quickly about the model's inner workings, which is not beneficial to the goal of the explanations. This fast thinking is not helpful for the analysis needed, and the end-user should be spending time trying to understand it, and thinking fast might be counter-productive to understand the graphs fully.

Although model-agnostic explainability techniques separate the interpretation process from the model itself, allowing for flexibility and comparison across models, they are highly dependent on high accuracy. If the machine learning model does not achieve high accuracy, then the conclusions achieved by the explainability techniques might be different from those of a more accurate model. Explainability techniques are therefore only good as an estimation. In Liu et al.'s paper, where they compare the explanations produced by two separate models with similar accuracies [22], they found that the next best model that their autoML system chose gave slightly different explanations. The best performing model was a stacked ensemble of random forests. After the stacked ensemble, a random forest was second-highest and marginally less accurate. Although the difference between the two accuracies was only marginal, there was some variance in the results from the explainability techniques. In the random forest, the ordering of the permuted feature importance and the extent to which features interacted through the H-statistic differed from the stacked random forest. Although explainability techniques sometimes attempt to explain tell quite similar processes, they do not achieve their explanations in the same manner. It is for this reason that many techniques are needed to get a fuller picture.

2.5. Natural language generation

The need for a more human-centred approach to both autoML and explainability leads naturally to the field of natural language generation (NLG). The area shows promise as explanations can be more explicit than how they currently are given and could offer more convincing explanations if they are done in natural language. In light of Kaur et al.'s [19] finding that data scientists over-trust and misuse interpretability techniques, it is timely that a rethink on how to do things better is in order. NLG is a research field that is situated in artificial intelligence as well as computational linguistics that attempts to produce output in natural language from a wide variety of different inputs. The output of NLG systems is always text but the input can be a wide range of things such as knowledge bases, images, graphs or plain text. Broadly speaking, they can be separated into two categories (a) text from data, or (b) text from text. Types of NLG systems range from template-based, rule-based and more contemporary encoder-decoder neural models. Examples of NLG systems can be as diverse as generated narratives about birds and their migration patterns [38], or summary of a baby's medical state when in an intensive care unit [31].

There are many approaches on how to build NLG systems, but there is no shared approach. Pipelines can be helpful for converting the language generation task to various sub-problems. In end-to-end NLG, such as what this thesis is focused on, the traditional pipeline proposed by Reiter and Dale [34] is less fixed and is more dynamic in structure than "end-to-end" systems. End-to-end systems have a more stochastic approach to development. There is less of a focus less on fixed structure and more on what is suited to the particular constraints of the inputs and desired outputs. End-to-end systems are a little more domain-specific than other forms of NLG.

Research has found that natural language generation systems can help decision making processes from uncertain data sources when compared to graphical representations of data [13]. Gkatzia et al. also showed that there were significant differences between genders and that women much

preferred the text-only condition of their experiment. Interestingly, men were more likely to be confident in their decisions when presented with only graphics or a multimodal representation of the data. Confidence was lowest in the NLG condition overall, but the NLG condition led to significantly better decision making than the graphical representations. Elsewhere, work has found that people can be better informed through generated text than graphical data representation in the medical domain [21, 40]. Law et al. focused on whether textual summaries of patient information might help inform doctors and nurses at varying levels of expertise about the patient's condition and asked how to proceed. The information was displayed as a) a trend graph on a screen or b) textual summaries that described the patients' condition. Both modes of explanation provided descriptions of the changing values of the physiological parameters of the patient and any relevant medical interventions without any level of medical interpretation. The results found that participants selected more appropriate actions when presented with textual information than graphs. In another study [40], Van Der Meulen et al. also found their participants preferred generated text over graphical representations of data. The study focused on the neonatal intensive care domain and found results that echoed Law et al.'s findings. Interestingly, in the study, the participants were familiar, and well practised with the graphical representations of the data but still performed worse when using graphs.

2.6. Natural language explanations

There seems to be very little research done on natural language explanations. However, they must be accurate, functional and easy to comprehend. The challenges in making good natural language explanations for AI systems are the same challenges that face natural language generation practitioners more generally. The explanations need to be adapted for specific purposes and users, contain narrative structure, communicate uncertainty and need effective evaluation [33].

In the development of intelligent systems, it seems like an essential requirement that these systems can explain their actions and decisions [20]. It matches intuition that a critical ingredient of explanations is for them to consider who the end-user is. A human would change their explanations whether they talked to a child or a professor. Part of the need for explaining to the user is the push for making explanations in some way plausible to the user, who can then make better decisions from them. With these in mind, explanations in natural language should be a prominent characteristic to communicate effectively.

One of the fundamental difficulties for this project, is that in one of the few papers on natural language explanations, Reiter argues that explanations should be written for a specific purpose [33]. Although users of autoML systems could be using explainability systems for any number of reasons, the one common purpose that they have is that they're looking to understand the underlying models and processes that are happening under the hood of said system. If natural language explanations are to be done for an autoML system, it is hard to make the language general enough so that if the user changes the dataset, then the language is both (a) specific enough to be informative while remaining (b) broad enough to generalise once another dataset is used. The natural language explanations were primarily template-based in structure and require human evaluation to test that the language of the explanations was clear and appropriate.

2.7. Multimodal explanations

Research has been done to offer multimodal explanations to justify image classification decisions of neural networks [30]. In the research, natural language justifications of decisions and a heat map highlighting the part of the image that led to the decision is given. The models provided both textual explanations and visual explanations for the image classification. Their research found that the two modes of explanations together can be more informative than unimodal explanations. This work asks the question more generally of whether users understand data in a textual format better than a visual format.

Although Park et al. compare graphical information and textual information combined as well as both modes individually [30], they conclude that combining text and graphs are more informative than either alone. However, they also conclude that at times their participants got more information from individual modes of information in particular cases. This is what Van Der Meulen and

Law et al., who found generated text to be more informative than graphical in both cases [40, 21]. Therefore, it is perhaps counterproductive to rule out multimodal explanations. It seems a viable secondary research question within this study. Two modalities combined could provide complementary explanations to one another. A design for the user study therefore is that there will be three conditions: a) one for graph explanations, b) one for textual explanations, c) one combining graph and text.

2.8. Evaluation

Evaluation of Generated Natural Language is a complex problem. It is difficult as there are many different ways of carrying out an evaluation, and one of the main issues is that automatic evaluation metrics do not work as well as they do in other NLP contexts. Part of the problem is that the evaluation methods do not correlate with one another and often give very different results [11]. A small, preliminary study during the development of textual explanations is fundamental. In NLG there are intrinsic and extrinsic methods of evaluation. In intrinsic methods, there are subjective human judgements and judgements involving human corpora. They might focus on aspects such as fluency or readability and accuracy, adequacy, relevance or correctness. In contrast, extrinsic methods of evaluation measure the effectiveness the text achieves a desired goal. Extrinsic methods lie further on the objective side of evaluation. The effectiveness depends largely on the application space that the text is designed for. In the case of this research, the preliminary pilot study is intrinsic in style and the main study is an extrinsic method of evaluation. The preliminary pilot is focused on whether the text reflects what is in the graphs and is subjective about the relevance, correctness and accuracy of the text. The main study on the other hand is focused on the effectiveness of the explanation where users are tested on the understanding of explanations. In autoML, researchers have used user studies to evaluate interpretability of the autoML model selection process [28]. In NLG, some have called for using only human assessment, but standardised methods are needed. Howcroft et al. go so far as to say that NLG evaluation over the last twenty years has been confused, and evaluation to this point has been notoriously complex [18]. Howcroft et al. propose a list of reporting recommendations for human evaluations, which they think are the minimum of what is worth including in reports of human evaluations. This checklist is a general guideline for structuring the generated text and can inform design of the text as well as the points to test in the preliminary and the main study. This gives a guideline to providing adequate text.

2.8.1. Evaluation checklist

- System
 - What problem are you trying to solve?
 - What do you feed in and get out of your system?
- Evaluation Criteria
 - What is the name for the quality criterion you are measuring?
 - How is the quality criterion defined?
- Operationalisation
 - How are you collecting responses?
 - Are your participants responding to?

After the preliminary study, we conducted a more comprehensive user study to assess the generated textual explanations' quality against the graphical explanations. It was not only necessary that the explanations themselves were evaluated, but that the end user's understanding of the explanation was evaluated [1]. The primary goal of this research is to test whether the generated text is more informative than the visual explanation. There is a secondary goal of seeing if the generated text and visualisation combined are more informative overall. The main issue with human evaluations is that they take more time and sometimes require domain expertise to apply to a broader

context. Experimental design has a massive effect on the quality of the evaluation. An explanation is only considered a good one if people find it helpful in the specific context [20]. Therefore selecting the appropriate participants was critical in the evaluation of this project. They therefore needed machine learning or autoML experience as much as possible. Unfortunately, running an autoML system for each participant is not be feasible as it takes a long time to run. Instead a simplified version of a likely scenario that a user might encounter where they have to evaluate explanations was the setup instead. This setup is modelled on Doshi-Velez et al.'s guideline for explainability evaluations of *real humans, simplified tasks* [7].

Chapter 3

Methods

3.1. Implementation

This section will describe the data, models, explainability techniques, and natural language generation modules. We performed one pilot study to assist with the development process and performed our main study to test the research question. Initially, the cervical cancer dataset was used to build the first steps of the language generation module. Then, the Australian weather dataset was added to test and ensure good generalisability beyond the cervical cancer dataset and variables. We later added a third dataset from the national basketball association (NBA) to test the generation modules for further generalisability. The third dataset was added late in the development process after the preliminary pilot study. For readability, all three datasets will be discussed first in the same section, followed by the classification models. The sections are as follows: implementation of the language generation modules, the preliminary pilot study that followed and the amendments after the pilot study. Instead of running an entire AutoML system, we focused on the part of the AutoML process that was relevant to act as an abstraction scenario focusing on explaining the outputs.

3.2. Data

Three data sets were selected for developing the modules. These data sets were selected based on ease of understanding of what was being predicted. We wanted participants to understand what was being predicted without difficulty so they could focus on the explanations. The feature variables in the datasets were not difficult to understand for the users and would easily allow participants to focus on the more complicated parts of what was being asked in both the preliminary study and the main study. This tactic was vital as we hypothesised that many of the survey participants might have limited experience with XAI techniques or machine learning. The three data sets were tabular with categorical and mostly numerical variables.

3.2.1. Cervical cancer

The first dataset used was the risk factor for cervical cancer dataset and predicts whether someone is at risk of having cervical cancer. The dataset was taken from 'Hospital Universitario de Caracas' in Caracas, Venezuela. The dataset was chosen as it was recommended by Christoph Molnar in his book on Interpretable Machine Learning [25]. Cervical cancer is a significant cause of mortality in many countries, and it can be prevented easily with effective screening processes. In the Republic of Ireland, there has been a recent controversy where a recent scandal led to many patients being classified as false negatives. False negatives are much more damaging than false positives in the medical domain (recall is more important than precision). Some 208 women affected by the controversy have since died. As a result, there has been a significant lack of trust among women in the screening programme since [3].

The dataset consists of data from 858 patients with 33 variables of booleans and integers. The variables contained are demographic information, habits and historical medical records. Several patients decided not to answer some questions due to privacy concerns, indicating a slightly biased dataset. The target variable is a biopsy, which identifies whether the person needs to undergo a medical procedure to test for cervical cancer. A biopsy is a gold standard for screening strategy for detecting pre-cancerous cervical abnormalities [10]. Random oversampling of the minority class

Dataset	Num Variables	Instances	Type of target
Cervical Cancer	33	858	Biopsy (binary)
Australian Rain	22	145,460	Rain Tomorrow (binary)
NBA (Basketball)	20	1,340	Career (binary)

Table 3.1: Dataset variables, instances and target type. Note: NBA target is a whether the player had a career of greater than 5 years past rookie season

Dataset	RF Accuracy	GB Accuracy	ADAB Accuracy
Cervical Cancer	0.98	0.99	0.82
Australian Rain	0.95	0.95	0.78
NBA (Basketball)	0.80	0.80	0.69

Table 3.2: Random forest, gradient boosting and Ada boost on all three datasets. Difference between random forest and gradient boosting classifier is quite small.

was performed as only 5% of records contained a true value for biopsy.

3.2.2. Australian weather data

The Australian weather data set is from the Australian Government's Bureau of meteorology [24]. The dataset contains ten years of daily weather observations from all over Australia. The target variable is *RainTomorrow*, which has the value of 1 if there was rain recorded above one millimetre and zero if otherwise. The variables are primarily numerical with some categorical, which totalled thirty-three overall. There are 145,460 instances in the dataset. The variables include pressure, temperatures, humidity, sunshine, location, and other relevant environmental readings. Predicting rain weather patterns is becoming more complicated in recent years with more variability due to climate change. This variability can have severe effects, particularly on areas already prone to drought and flooding.

3.2.3. National basketball association

The national basketball association of America (NBA) releases data on its players for public use. The dataset consists of player statistics on games played. The dataset was compiled to predict rookie player career length and whether players will last five years passed their rookie season [44]. Accurately predicting the probability of a player lasting longer than five years can benefit team management and the players themselves.

Unlike the other two datasets, which had reasonably obvious variable names, this dataset had variable names that probably wouldn't be transparent to the novice reader. The original letters were codes. "GP", "FGA", and "TOV" were supposed to signify "Games played rookie season", "Field goals attempted per game", and "Average turnovers per game". This dataset, therefore, was the only dataset where we changed the variable names from their original.

There were 1,340 instances and 20 variables. Average free throws per game, average games played, and games played during the rookie season are some variables in the dataset. The player name was taken out of the dataset as it was an unnecessary variable. As most players did not last longer than five years, random oversampling of the minority class was used to bring both classes to a balance.

3.3. Classification Models

To test whether natural language explanations offer more utility than the standard visual methods, we trained classification models on three different data sets to test the language generation module. The algorithms used were gradient boosting trees, random forest, support vector machines, neural networks and AdaBoost. Random forest, gradient boosting trees and ADABOOST all achieved high accuracy. Neural networks seemed to require too much data, and two out of the three datasets are quite small. Only an accuracy of 55% was obtained with the neural network on the cancer dataset. Support vector machines did not work very well either, only achieving an accuracy of 65%. Logistic regression was also used, but it did not prove easy to get a worthwhile accuracy. Only an accuracy of 67% was gotten. Perhaps higher could be done by performing complex feature engineering. Support vector machines, neural networks and linear regression were only tested on the cervical cancer dataset and because they didn't perform highly were not considered any further.

At the start of the project, the highest-performing model for each dataset was used to produce explanations. Although this seemed the best thing to do, we used the same model for all three

Dataset	Accuracy	Precision	Recall
Cervical Cancer	0.99	0.99	0.99
Australian Rain	0.95	0.95	0.95
NBA (Basketball)	0.81	0.81	0.81

Table 3.3: GB accuracy, precision and recall

datasets instead. This choice was made to avoid testing differences in models rather than differences in explanations. The gradient boosting classifier was selected in the end. There was very little difference in accuracy between the random forest and the gradient boosting classifier, so either could have been used. However, because we decided to use the same model, a compromise was made regarding very high accuracy on all three datasets with the NBA dataset being the lowest out of the three. It did not seem possible to achieve an accuracy much higher than what was possible with the gradient boosting classifier (0.80). Adaboost performed quite badly on the Australian Rain dataset as well as the NBA dataset so it was not considered any further.

After a high accuracy was achieved on all three data sets, permutation feature importance, partial dependence plots and accumulated local effects plots were run post-hoc.

3.4. Natural language generation module

For the natural language generation module, we wanted to express what the most important parts of each of the graphs were. We felt that NLG allows us to be more explicit about certain elements and leave out any parts that might be worth overlooking. In designing the NLG module, we needed to plan whether to opt for a rule-based system, template-based system or neural system. Some argue that the distinction between the template-based systems and rule-based systems is outdated and that there are more similarities between the two than previously given credit for [6]. Despite template systems being more difficult to maintain than a more complex linguistic syntactic processing method, their simplicity to implement is their key advantage. We opted for a template-based system for this reason. We thought that a neural approach wouldn't suit the nature of the problem because of a lack of data to train on. Another issue is the lack of specificity and control that comes with them. The templates for partial dependence and accumulated local effects were largely written based on Christoph Molnar's examples [25]. These were used as a starting point for the templates' design.

3.4.1. Permutation Feature Importance

Traditionally, permutation feature importance outputs a boxplot with variables on the x-axis and mean importance on the y-axis. PFI also gives the standard deviation. Some consideration was given to whether or not to include standard deviation. In the end, we decided to leave it out as it was not included in the graphical example of PFI given by Molnar with a textual description [25], see figure 3.3. We also decided it might be best not to overload users with too much information that we thought might not be worth focusing on.

There are three main categories of variables displayed in permutation feature importance. After permutation, there are (a) variables that make the accuracy of the model decrease (*important*), (b) variables that do not change the model's accuracy (*unimportant*) and (c) variables that lead to an increase in the model's accuracy (*negative importance*). These three distinctions for groups were the starting point for producing language categories of language templates. From these three categories of templates there would be modifications based on how many variables there were. The variables were also formatted to appear in quotation marks with commas in between. There were also sentences describing their role in the model. There is an example of the generated text along with the associated graph in figures 3.4, 3.5. An example of the templates for these is below.

3.4.1.1. Template for permutation feature importance

[There] [are / is] [*number important variables*] [important variable/s]. [Removing one of these variables individually will lead to a decrease in the model's accuracy.] [The variable/s that are important are][*list important variables*]

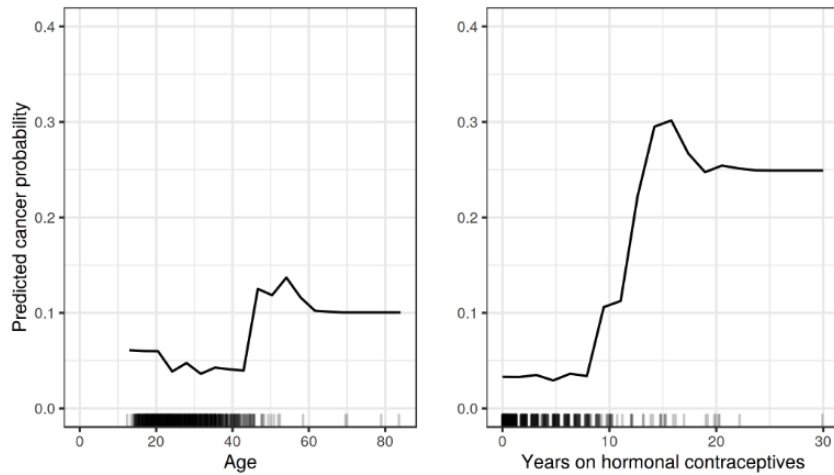


FIGURE 8.3: PDPs of cancer probability based on age and years with hormonal contraceptives. For age, the PDP shows that the probability is low until 40 and increases after. The more years on hormonal contraceptives the higher the predicted cancer risk, especially after 10 years. For both features not many data points with large values were available, so the PD estimates are less reliable in those regions.

Figure 3.1: Christoph Molnar PDP example with text explanation [25]

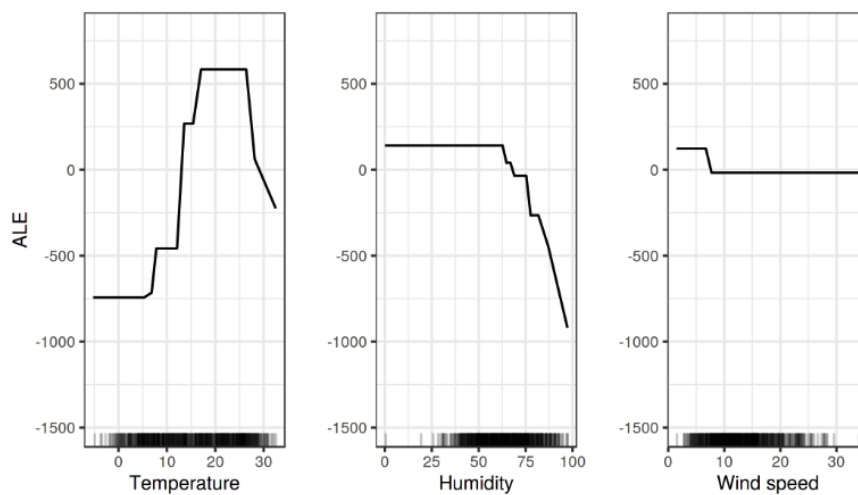


FIGURE 8.11: ALE plots for the bike prediction model by temperature, humidity and wind speed. The temperature has a strong effect on the prediction. The average prediction rises with increasing temperature, but falls again above 25 degrees Celsius. Humidity has a negative effect: When above 60%, the higher the relative humidity, the lower the prediction. The wind speed does not affect the predictions much.

Figure 3.2: Christoph Molnar ALE example with text explanation [25]

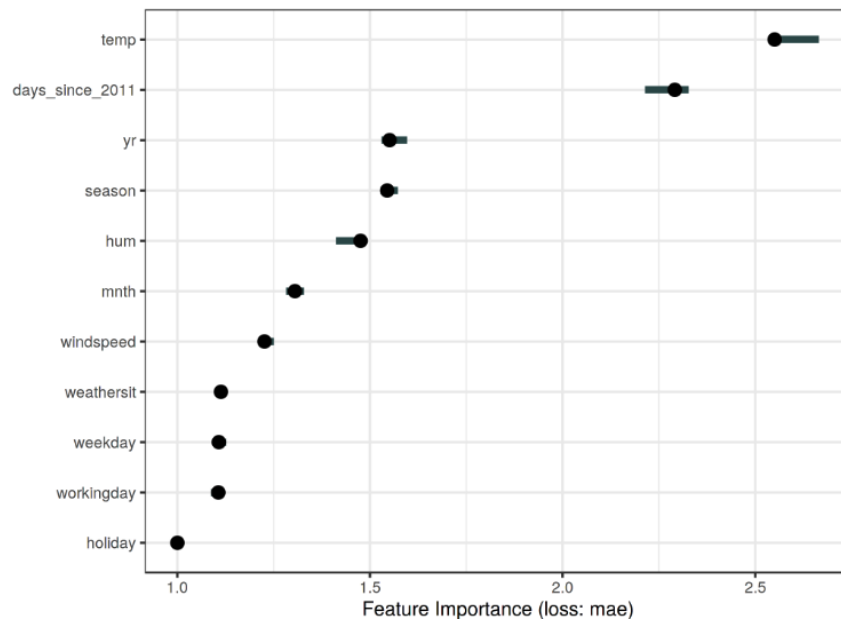


FIGURE 8.27: The importance for each of the features in predicting bike counts with a support vector machine. The most important feature was temp, the least important was holiday.

Figure 3.3: Christoph Molnar PFI example with text explanation [25]

[There] [are / is] [*number unimportant variables*] [variables that don't affect the model's accuracy.]
 [The variables that are not important are] [*list unimportant variables*]
 [There] [are / is] [*number negative important variables*] [variables that impact the model's accuracy negatively.] [Removing them individually could lead to an increase in overall accuracy.] [The variable/s that have a negative influence on the prediction are] [*list negative important variables*]

3.4.2. Partial Dependence and Accumulated Local Effects

Other than some differences in the wording produced for each, the process used to produce language was almost identical for partial dependence and accumulated local effects. Therefore, these techniques will be covered together as the main parts of the language generation module were essentially the same. The output obtained from the two techniques was the raw data from each one of the graphs. For partial dependence and accumulated local effects it was an array of the x and y values from the graph. Content determination in partial dependence and accumulated local effects was not straightforward. At the time of writing, there is no known knowledge base or dataset with example explanations. Yu et al. had a similar problem: they produced summaries of time series data for a gas turbine [43]. Their research had the advantage of having a previously-built knowledge base. Below is the template used for both partial dependence and accumulated local effects. Much of the design was taken from 3.1

3.4.2.1. Template for partial dependence and accumulated local effects

Refer to figures 3.6 and 3.7 when observing the example template below.

[The variable

[variable name] [has a direct effect on the outcome]] / [[The variable] [variable name] [has no effect on the outcome]]

[Overall the average marginal effect of the model predicting] "[variable name]" [is between] [minimum y value] [and] [maximum x value].

[The probability of the model predicting] ["variable name"] [starts at] [first y value] [then as] ["variable name"] [goes up, the probability] *loop over values in LOWESS coefficients*: [increases [after x

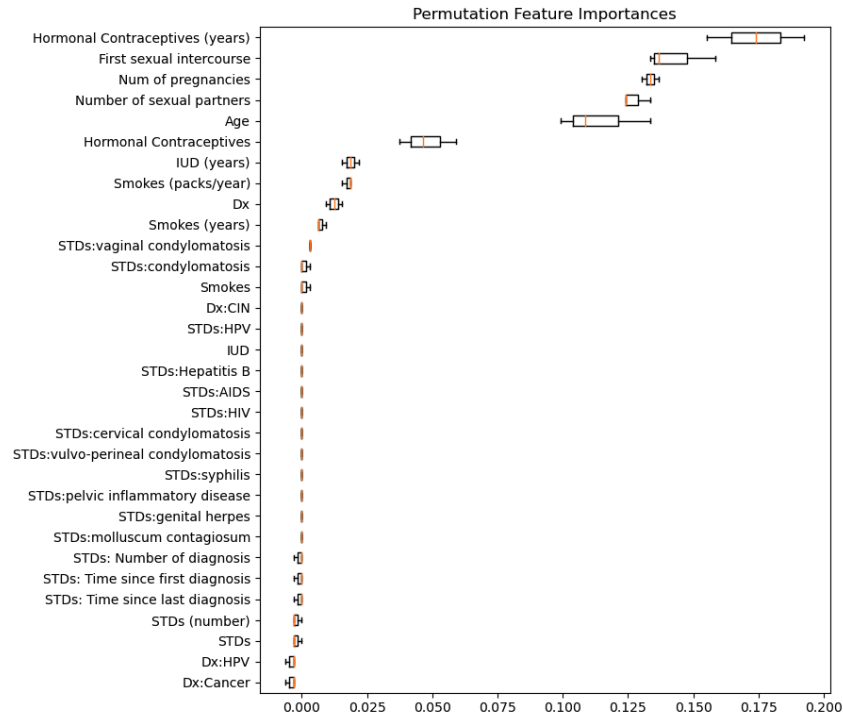


Figure 3.4: Implementation 1 visual explanation (PFI Cancer)

Generated Textual Explanation

There are 13 important variables.

Removing one of these variables individually will lead to a decrease in the model's accuracy.

The variables that are important are "Hormonal Contraceptives (years)", "First sexual intercourse", "Num of pregnancies", "Number of sexual partners", "Age", "Hormonal Contraceptives", "IUD (years)", "Smokes (packs/year)", "Dx", "Smokes (years)", "STDs:vaginal condylomatosis", "STDs:condylomatosis", "Smokes".

There are 12 variables that don't affect the model's accuracy.

The variables that are not important are "IUD", "Dx:CIN", "STDs:HPV", "STDs:Hepatitis B", "STDs:HIV", "STDs:AIDS", "STDs:molluscum contagiosum", "STDs:genital herpes", "STDs:pelvic inflammatory disease", "STDs:syphilis", "STDs:cervical condylomatosis", "STDs:vulvo-perineal condylomatosis".

There are 7 variables that impact the model's accuracy negatively.

Removing them individually could lead to an increase in overall accuracy.

The variables that have a negative influence on the prediction are "STDs: Number of diagnosis", "STDs: Time since first diagnosis", "STDs: Time since last diagnosis", "STDs (number)", "STDs", "Dx:Cancer", "Dx:HPV".

Figure 3.5: Implementation 1 textual explanation (PFI Cancer)

Visual Explanation

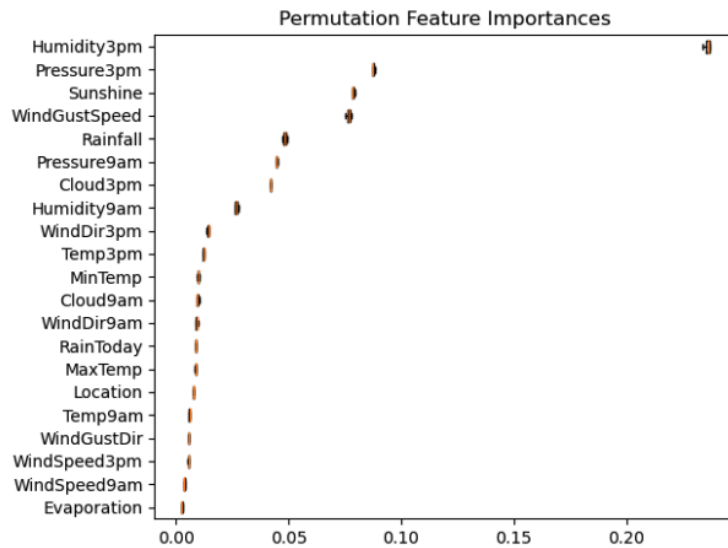


Figure 3.6: Implementation 1 visual explanation (PFI Rain)

Generated Textual Explanation

There are 21 important variables.

Removing one of these variables individually will lead to a decrease in the model's accuracy.

The variables that are important are "Humidity3pm", "Pressure3pm", "Sunshine", "WindGustSpeed", "Rainfall", "Pressure9am", "Cloud3pm", "Humidity9am", "WindDir3pm", "Temp3pm", "MinTemp", "Cloud9am", "WindDir9am", "RainToday", "MaxTemp", "Location", "Temp9am", "WindGustDir", "WindSpeed3pm", "WindSpeed9am", "Evaporation".

There are 0 variables that don't affect the model's accuracy.

There are 0 variables that impact the model's accuracy negatively.

Figure 3.7: Implementation 1 textual explanation (PFI Rain)

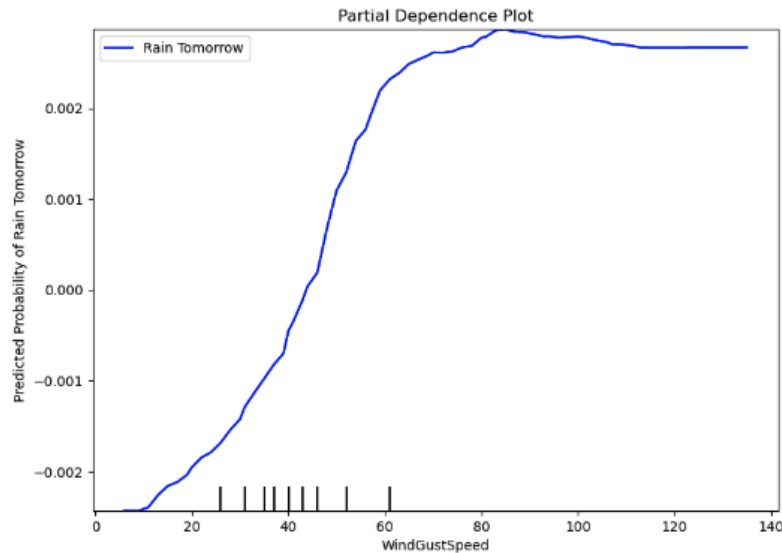


Figure 3.8: Implementation 1 visual explanation (PDP Rain)

location] / decreases [after x location / stays constant]]

The first thing to produce language from was whether the feature variable affected the model's outcome. If values changed along the y-axis, a sentence would be outputted. From this change, two sentences were constructed to be produced if (a) the feature variable had a direct effect on the outcome ("the feature (x) directly affects the model's outcome") and (b) if the feature variable had no effect on the outcome ("There is very little change in the model's prediction based on changes in the feature (x)"). Another sentence was constructed to signify the range of values that the feature effect had on the outcome. These were taken from the maximum and minimum values of the line plot on the y axis. The following sentence that is outputted is: "Overall, the average marginal effect of the model predicting "Cervical Cancer" based on "Num of pregnancies" is between 0.33142 and 0.43551." This sentence takes the range of values along the y axis and the variable names.

One of the problems we had when designing the language was the variability of inputs. Sometimes there would be many peaks and troughs along the line plot, sometimes very few. This variability made it difficult to decide what to output. Each time the feature of interest or the dataset is changed, the required explanation must accommodate that, and sometimes changes were drastic. ALE and PDP have dissimilar shapes to their lines to further complicate things. ALE has much more constant, flat lines where the y axis does not change in value. We decided that describing increases and decreases in the effect based on feature value was the most important thing to consider but we needed a way of aggregating this down to a manageable level. This language was less straightforward to implement than the other sentences constructed and needed an extra process to do so.

3.4.3. Local Regression

We decided we needed a way of logically and automatically distilling down the information to a simpler format to produce language from. We decided that Local regression might be a way to aggregate the complex information that was displayed into a more digestible format, simplifying the main parts of the graph which would in turn operate as a basis to produce sentences from. This would help with the sentence aggregation step. Local regression fits many lines to a curve. It is a good option when a regular linear regression does not provide enough fidelity. A difficulty of the accumulated local effects plots was that each time the output would be a different length in values and caused errors when attempting to load outputs from the technique into the LOWESS regression step. Some error catching alleviated this whenever it was run. This error handling step

Generated Textual Description

The variable "WindGustSpeed" has a direct effect on the outcome.

Overall, the average marginal effect of the model predicting "RainTomorrow" based on "WindGustSpeed" is between -0.0025 and 0.00286.

The probability of the model predicting "RainTomorrow" starts at 0 then as "WindGustSpeed" goes up, the probability increases after 28, then stays constant.

Figure 3.9: Implementation 1 text explanation (PDP Rain)

significantly slows down the process for accumulated local effects.

The local regression was Locally Weighted Scatterplot Smoothing technique (LOWESS). This handled the outputs of both techniques. We also tried using Locally Estimated Scatterplot Smoothing (LOESS), but the results were much more accurate with LOWESS. We assumed this was because small amounts of data were used to fit the curve. LOWESS works much better with smaller amounts of data than LOESS.

Each locally weighted regression line's slope and intercept coefficients are then kept. We decided that choosing a small number of fits in the hyperparameters of the local regression would allow us to get a rough estimation of the process behind the line plot and was a good way of summarising the many peaks and troughs that occurred. A high R-squared value was obtained each time. This R-squared value was achieved with a looping process over the other parameters.

The slope coefficients of each regression line were used to identify where the increases and decreases were. The intercepts were used to find the values along the y axis. A loop was then used to iterate over all slopes and produce a sentence. For example, the output from this step alone would be something like, "The effect increases, then decreases, then increases, then decreases..." (see figure, last paragraph 3.11)

Early in the process, we used the values from the slope coefficients not only as signifiers of increases or decreases but as a modifier to describe the rate of increases or decreases. An example might look similar to 3.10 with generated text 3.5. Instead of the example reading "... increases after 0, then increases after 3 *by a lot*, then decreases after 6 then stays constant." The "by a lot" and "a little" were modifiers related to the sharpness of the increase or decrease. We set a threshold of a 60-degree angle or more for a significant increase. Less than 30 degrees was used as an increase or decrease of a little. Nothing would happen if the angle were in between these values. This modifier worked well sometimes, but it seemed entirely out of place when the sharp increases were very short in duration and could lead to confusion if the "increase by a lot" only related to less than 5% of the line. Eventually, we decided not to include it.

One of the main issues when designing the language generation module for the outputs of varying lengths was the difficulty in rounding numbers. We attempted to do a rough rounding of only two numbers behind the decimal place, but doing this could lead to scenarios where the language would have comparisons between 0.00 and 0.00 when numbers were very small and fell within a narrow range. For example, the numbers could be as low as 0.0000001 or lower without any rounding. This unedited output seemed a little too robotic, unlike what a person would say. We roughly set a threshold of two digits after the decimal place and chose variables that would not output numbers that were too small.

3.5. Evaluation

3.5.1. Preliminary pilot study

To help in the development process in deciding what to include in the generated text, we conducted a preliminary pilot study to evaluate what had been done so far. This pilot study was done to see whether the generated text reflected what was shown in the graphs.

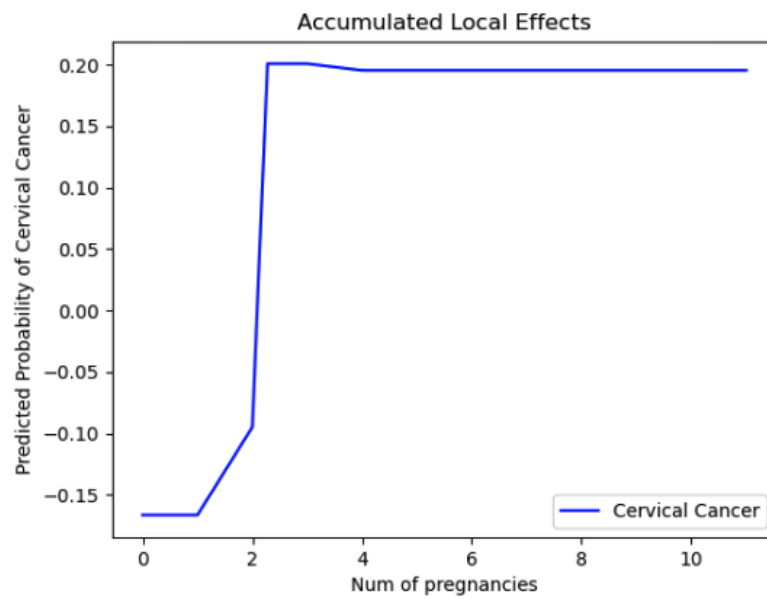


Figure 3.10: Implementation 1 visual explanation (ALE Cancer)

Textual Explanation

The variable "Num of pregnancies" has a direct effect on the outcome.

Overall, the average marginal effect of the model predicting "Cervical Cancer" based on "Num of pregnancies" is between -0.19529 and 0.22106.

The probability of the model predicting "Cervical Cancer" starts at -0.2 then as "Num of pregnancies" goes up, the probability increases after 0, then increases after 3, then decreases after 6, then stays constant.

Figure 3.11: Implementation 1 text explanation (ALE Cancer)

Does the text reflect what is in the graph? *

1 2 3 4 5

No Yes

Would you change anything if you were to rewrite it?

Your answer _____

Figure 3.12: Preliminary pilot study question example

3.5.1.1. Participants

Eleven people participated in the experiment. We presumed that the survey was most applicable to people with a background in machine learning. The participants were selected based on this requirement. Nine participants are currently obtaining or have completed an MSc in Artificial intelligence. Another participant has an MSc in Data Analytics. Many of the above work in data science, artificial intelligence or software engineering. One is studying a PhD in pharmacology who uses statistics and modelling extensively. Another is a business consultant who uses machine learning in their work. The mean experience level was gathered via a five-point Likert scale between one (novice) and five (proficient). All the participants have some background in machine learning. Some participants had experience with XAI, with a mean score of two on the Likert scale. All participants were European, and most were male. Each participant consented that (a). their data would be used for research purposes according to GDPR (b). information was gathered anonymously, (c). They knew they could terminate the survey at any time, (d) they were eighteen years of age or older. There could have been some sampling bias in the participant selection as each one of the participants knows the researcher conducting the project.

3.5.1.2. Design

The survey was a within-subjects design where all participants answered all questions, and there was only one version of the survey. The first of the two sections centred around the Cervical cancer dataset. The second was on the rain prediction dataset. Each section had one example of Permutation Importance, Partial Dependence Plots and Accumulated Local Effects plots. The examples consisted of a visual explanation along with the generated text. In total, there were three examples of explanations for each dataset. The features selected to be shown were simply chosen for their ease of understanding of what was being predicted. We did not want to distract from the language or graphs.

The participants were asked whether the text reflected what was in the graph. Their response was captured via a Likert scale between one and five. They were also asked if they would change anything in the text if they were to rewrite it. The results section below will first describe the feedback on the quality before summarising some suggestions on how the participants would rewrite the text.

One participant was given the survey before sending it out to all other participants to test whether the instructions and survey made sense. There were no issues after this pretest. Therefore, all participants sent the survey, which was the same as the pretest. Because there were no issues with the pretest, answers recorded in the pretest were included in the final results.

	Cervical Cancer			Rain			Mean per technique
	Mean	Std	Range	Mean	Std	Range	
PFI	4	1	2,5	3	1.41	1,5	4
PDP	4	1	2,5	3	1.41	1,5	3
ALE	3	1.41	1,5	4	1	1,5	4
Totals	4			3			4

Table 3.4: Preliminary pilot study results. Mean quality ratings for Permutation Feature Importance (PFI), Partial Dependence Plots (PDP) and Accumulated Local Effects Plots (ALE). All averages were calculated before numbers were rounded. Last column is mean rating per each technique across both datasets

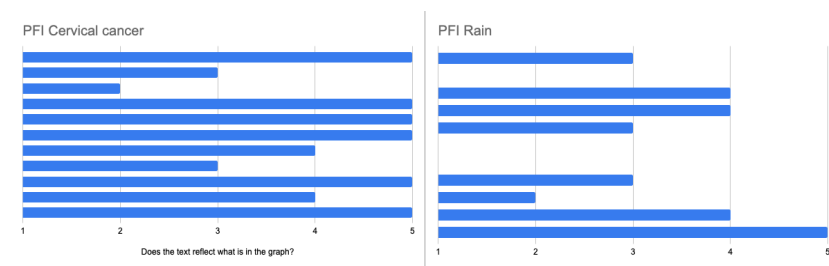


Figure 3.13: PFI scores likert scale. Y-axis represents each participant, x-axis represents score from one to five. The question was "Does the text reflect what was in the graph?"

3.6. Results of preliminary pilot study

Because there was a small number of participants in this pilot study, there was high variance in responses 3.6.1.. However, as this was a preliminary study looking for suggestions and a guide during development, a small number of participants is sufficient. The results section consists of a summary of the first question asked about the generated text's accuracy to the graph. Then, the second part will focus on summarising people's suggestions for how they would rewrite the text.

3.6.1. Quality

This section is related to the mean quality of ratings in table 3.4. The mean of all techniques over both datasets is 4. 5 is the maximum quality score. This finding shows us that people's opinion of the text was more positive than negative overall. Generating natural language explanations is an emerging area with very little research, so this is a decent start. However, there seems to be a little room for improvement. It can be seen from the results that all three techniques have some positive and negative results. The partial dependence plots perform the least well out of the three techniques, with a midranking score of just three. This finding signifies that the generated text does not describe the visual explanation well. However, a three does not signify that the generated text is unsuccessful either. The other two techniques perform the same with a mean quality of 4 each. There are some differences between the quality of the generated text from the rain dataset and the cervical cancer dataset, but these vary across different explainability techniques. The partial dependence plots and permutation feature importance perform better on the cervical cancer dataset, but the accumulated local effects plot performs better on the rain dataset. There was more variance in the rain dataset than cervical cancer, with permutation feature importance and partial dependence. Nobody gave the lowest quality rating on the cervical cancer dataset for permutation feature importance and partial dependence.

As can be seen from the boxplot of each technique 3.16. On the cervical cancer dataset, PFI and PDP did similarly with ALE doing slightly worse with a higher spread. The Rain dataset didn't do as well with PDP doing the worst, but ALE did the best overall.

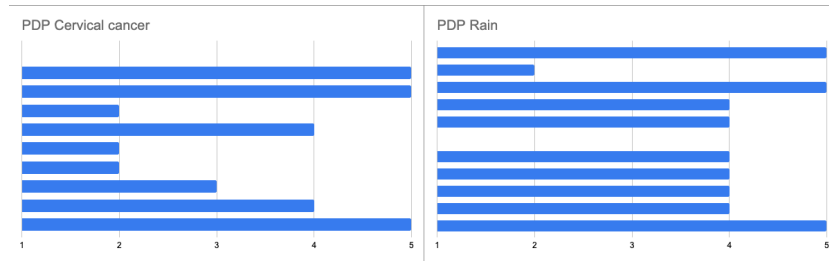


Figure 3.14: PDP scores likert scale. Y-axis represents each participant, x-axis represents score from one to five. The question was "Does the text reflect what was in the graph?"

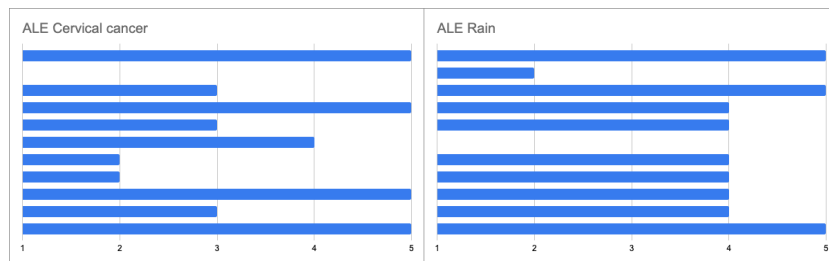


Figure 3.15: ALE scores likert scale. Y-axis represents each participant, x-axis represents score from one to five. The question was "Does the text reflect what was in the graph?"

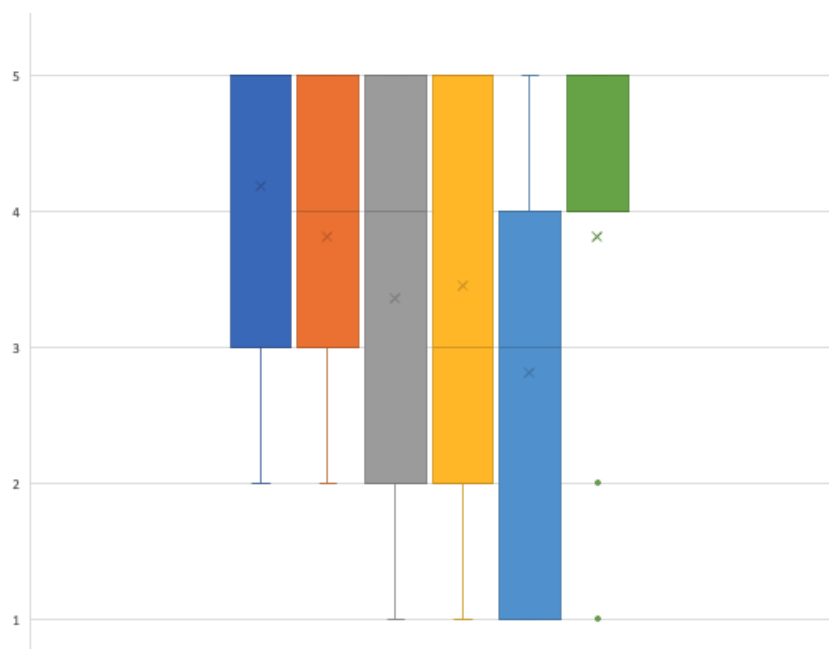


Figure 3.16: Scores for each explainability technique and dataset on likert scale. From left to right: PFI (Cancer), PDP (Cancer), ALE (Cancer), PFI (Rain), PDP (Rain), ALE (Rain)

3.6.2. *Suggestions*

After participants were asked whether the text reflected what was in the graph, they were asked what they would change if they were to rewrite the generated text. They were given a text box where they could give their answers. As the sample size of participants in this preliminary pilot study was small, the suggestions of what to focus on seemed quite varied.

3.6.2.1. *Permutation Feature Importance*

The text generated from permutation feature importance got the least number of comments on how it could be improved. This finding is taken to be a good sign. Commenters seemed to call out for finer differentiation between groups of variables. Even if they all lent themselves to an accurate prediction, they were not weighted the same. Other commenters suggested that it is not necessary to list out all variables. Another commenter mentioned that the text should be removed entirely and that the graph is more explicitly specific because it shows how much each variable affects the accuracy. This comment was directly linked to the research question of whether explanations are better conveyed through natural language or visual formats. Most participants who said that the text reflected what was in the graph well (ratings of 4 or 5) did not suggest how the text could be improved. Most suggestions, therefore, are more negative than positive. This finding was expected since people were explicitly asked to write something only if there was anything they would change.

3.6.2.2. *Partial Dependence Plots*

One of the comments was that the text was repetitive, with the word "then" uttered too many times. Another user said that the black lines along the x-axis in the partial dependence plot were not mentioned. These lines were the data points that the partial dependence plots create, which come from averages between data points in the dataset. These points can be confusing as they were shown 2.25 "Num of pregnancies" in the cervical cancer dataset, which confused some participants. Some commenters mentioned that the text would work better if it only described the significant shifts in the graph. Two users mentioned that the fluctuations in the graph are not quantified in the text, highlighting that the increases or decreases could be interpreted as equal in magnitude.

3.6.2.3. *Accumulated Local Effects Plots*

The first two paragraphs did not get mentioned at all. Most were related to the descriptions of the increases and decreases produced mainly by the local regression. Participants wanted significant changes and not every detail. The specificity of the increases and decreases were not exact enough due to the local regression. Another suggestion was that ranges should be given between the increases and decreases. Repetition seemed to have been an issue for some. Another issue was that the numbers were too small to be used in a sentence, there was an issue with rounding. It should be noted that there was a mistake on the experimenter's behalf; something was overlooked in the experimental setup. The label on the y-axis read, "Predicted Probability of Rain Tomorrow" when it should have been "Accumulated Local Effects of predicted Rain Tomorrow".

3.6.2.4. *Summary of results*

The main issue with partial dependence plots and accumulated local effects plots that people seemed to comment on was that some of the values expressed in the generated text were not as exact as the visual explanation. The problem was with how the generated text was achieved. To recap how the text is generated from the graph, the values are taken from the partial dependence and then fed into a local regression model that returns values that the text generation module uses. The values directly from the partial dependence plot are difficult to interpret, so the local regression is used to allow for an aggregated, simplified interpretation. This interpretation is a generalisation due to the low number of fits set as a parameter on the LOWESS algorithm. The interpretation goes from the specific partial dependence data to a more general interpretation with the local regression and back to specific values in the generated text.

Interestingly, there seemed to be a little consensus between participants on which points exactly were the problem. People found the natural language explanations of the direction of the line

(increases and decreases and at what points) too vague or not exact enough. Weighting locations confused some participants when they were not expressed in the generated text but appeared in the graph. The generated text used weighting locations derived from partial dependence plots. Some people were confused by the values produced.

Most of the issues the survey participants seemed to have with partial dependence and accumulated local effects originated with the aggregation step. When the preliminary pilot study was run, the aggregation step occurred at the locally weighted linear regression, where selecting a small number of local regression fits aggregated the amount of generated text produced. This strategy led to a poor fitting line with a lower-than-optimal R-Squared value, which acted as the basis for the language generation step. There could be a better way of achieving this by moving the aggregation step further down the language generation module's pipeline. Regarding permutation feature importance, the prominent issue people seemed to have comes from grouping variables based on how much they affect the outcome. This finding only came up because of the inclusion of the rain dataset, which added a significantly more important variable than the rest, so this variable stuck out prominently.

3.7. Implementation Part 2

After the preliminary pilot study, it was clear that more fits needed to be added to the locally-weighted regression and choosing a small number of fits was a bad strategy. Choosing a small number of fits worked well for summarising the most notable increases and decreases in the graph but led to confusion once values were added to the text that was supposed to represent the actual values on the ALE or PDP. The module needed changing so that the aggregation step occurred further down the pipeline. This was because the local regression was definitely the source of many of the problems. We also improved the rounding step that occurred and changed the architecture to avoid repetition and describe macro-shifts that occurred in the graph rather than attempting to describe all details. In permutation feature importance, we changed the linguistic realisation as well as the text structuring through means of the added clustering step.

The preliminary pilot study successfully highlighted the issues with the text generation modules and gave us directions to pursue. One of the main changes that we decided to implement, was that we needed to group the important variables in a more logical fashion. This was achieved through applying a clustering step. We also decided that there needed to be some changes to language as well to increase fluency.

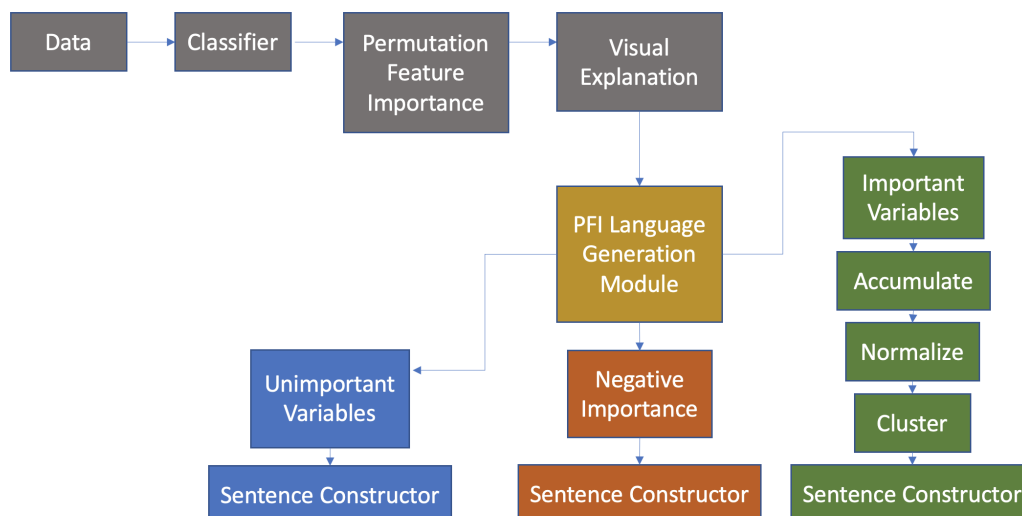


Figure 3.17: Permutation feature importance process graph version 2

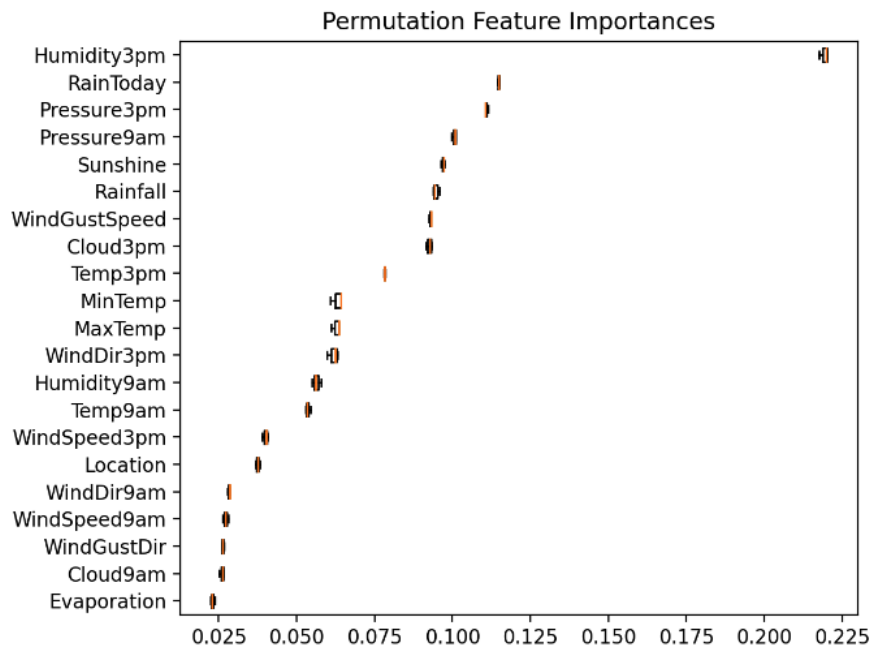


Figure 3.18: PFI rain visual

3.7.1. Permutation feature importance

3.7.1.1. New template for permutation feature importance

Below is the updated template for permutation feature importance. Refer to 3.18 and 3.19 as an example for the template. For ease of readability, checks for plurals will be left out of this description of templates. These lexicalisation changes are in the implementation, but it is much easier to read without them. Changes for quantities are used when there was one variable that was important, more or all. When there were zero variables, the code would do nothing and ignore that particular template.

[Permutation feature importance has revealed that] [there are] [number of variables] [variables]. [individually permuting these] [number of variables] [has lead to a decrease in the model's accuracy.] [The most important variables are] [list highest band of important variables]. [The variables] [list high band of important variables] [are also very important] [The variables] [list medium band of important variables] [positively affect the prediction, but not a lot.] [Other variables shown to have a positive effect on the model's accuracy were] [list lower important variables] [list lowest important variables]. [These variables affect the prediction, but only a tiny amount]

[number of unimportant variables] [variables don't affect the model's accuracy, either positively or negatively.] [the variables are] [list unimportant variables]

[number of negative important variables] [variables impact the model's accuracy negatively.] [Removing one of them could lead to the model's accuracy increasing] [The variables are] [list negative important variables]

3.7.1.2. Clustering

To group variables logically in permutation feature importance, we decided to accumulate the variables that positively influenced the model, that is, the variables that, after permutation, the accuracy went down. After accumulation, we normalised them. We then decided to divide three separate groups between 0 and 1 (under 0.33 was the first group, between 0.33 and 0.66 was the second, etc.). This step worked quite well, but when applied to other datasets, it did not work as well. Some

Permutation feature importance has revealed that all 21 variables are important. Individually permuting these variables has led to a decrease in the model's accuracy. The most important variables are "Humidity3pm", "Pressure3pm" and "RainToday". The variables "Sunshine", "WindGustSpeed" and "Pressure9am" are very important as well. The variables "Temp3pm", "Cloud3pm" and "Rainfall" have some positive effects on the prediction, but not a lot. Other variables that were shown to have a positive effect on the model's accuracy were "Humidity9am", "MinTemp", "Temp9am", "WindDir3pm", "MaxTemp", "WindDir9am", "WindSpeed3pm", "Location", "WindSpeed9am", "Cloud9am", "WindGustDir" and "Evaporation". These variables have some effect on the prediction, but only a very small amount.

Figure 3.19: PFI rain text

variables were grouped on the threshold border but were assigned as being different. We, therefore, needed a better tactic to group the variables after they were accumulated and normalised. We tested three algorithms: DBSCAN, affinity propagation and K-means clustering. Affinity propagation did not work well. For the cervical cancer dataset, where there were ten variables, Affinity propagation made ten clusters. DBSCAN worked much better, but the issue with it was that it was not fixed and had a varying number of clusters. We decided to use K-means instead as it had a fixed length in clusters. We decided that having a fixed number of clusters would be much more straightforward for the text structuring step. We used DBSCAN to find the number of clusters on the cervical cancer dataset, then set K to that number. We then tested this on the other two datasets.

3.7.1.3. *Changes to lexicalisation*

We felt some parts of the language could be left out if they did not add any real value. Instead of outputting the lines "There are 0 variables that don't affect the model's accuracy" and "There are 0 variables that impact the model's accuracy negatively", we decided to remove these entirely as there were 0 variables being talked about meaningfully. These lines seemed a little unnecessary and unnatural. We decided that there needed to be more fluid language so some particular focus was paid to the lexical choice and realisation. We added sentences and phrasing to make the text flow better and decided to fill gaps between the listed variables. We also changed all language to the past tense. Instead of "... removing it will lead to an increase in the model's accuracy", we put "removing it might lead to an increase in the model's accuracy". The difference in these two sentences is from it "will" to "it *might*". The vagueness is intentional because there is a stochastic nature to the model fitting process when considering gradient boosting trees or random forests. Therefore, removing the variable is not a guarantee that the model will achieve higher accuracy next time that it is run.

We also decided to account for cases that do not appear in the tested datasets or appear very sparsely. For example, we altered the language to account for cases with only one crucial variable or no important variables. We did this for unimportant, neutral variables and important variables. To avoid repetitive sentences, we included changes in sentence structure so that when a list of variables was mentioned within a cluster, the sentence would be different, not just the modifier. For example, "The variables x, y and z positively affect the prediction, but not a lot"... "Other variables shown to have a positive effect on the model's accuracy were p and q".

3.7.2. *Partial dependence and accumulated local effects*

3.7.2.1. *Templates for partial dependence and accumulated local effects*

An example of the partial dependence plot and accumulated local effects generation module templates are given below. This will be a simplified version, but an example might help the reader

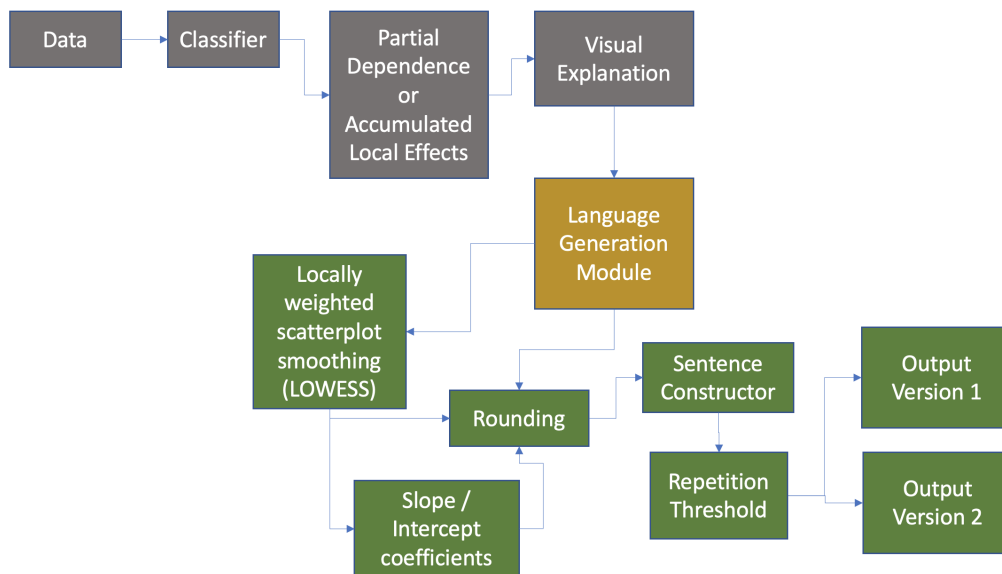


Figure 3.20: PD and ALE process graph version 2

understand better in figures 3.23, 3.24

[Overall the average marginal effect of the model predicting] [target variable] [based on] [feature name] [is between] [min y] [and] [max y]
 [There is very little change in the model's prediction based on changes in the feature] "[feature name]" / [The feature] "[feature name]" [directly affects the model's outcome]
 [The accumulated local effects of the feature / The partial dependence of the feature] "[feature name]" [on the model predicting] [target variable] [starts with an effect of] [start y] [when the feature's value is at] [start x] [then the effect]...

Version 1 [increases] [to an effect of] [new y] [when the feature's value is at] [new x] *repetition threshold + 1*,
 [decreases] [to an effect of] [new y] [when the feature's value is at] [new x] *repetition threshold + 1*,
 [stays constant] [when the feature's value is from] [beginning constant x] [to] [end constant x] *repetition threshold + 1*
[if repetition threshold exceeds a value of five the following templates will be used instead of the [increases...], [decreases...] and [stays constant...]]

Version 2 [There is a lot of variability in the output based on the feature variable] [The model is most likely to predict] [target variable] [based on the feature] [feature name] [when the feature is at] [x when y is at maximum value] [with an effect of] [maximum y value].
 [The model is least likely to predict] [target variable] [based on the feature] [feature name] [when the feature is at] [x when y is at minimum value] [with an effect of] [minimum y value].

One of the first things we did to alleviate some confusion was to be more explicit about changes in the graph. "the accumulated local effects of the feature (x) on the model predicting (y) starts with an effect of (position y) when the feature's value is at (x)". We included more values to be more explicit 3.22.

Another change we implemented after the preliminary pilot study was improving the part that handled outputting "stays constant". We increased the number of fits in the local regression and moved the aggregation step further down the pipeline nearer to the end where the language was outputted. This change that had aggregation happen later allowed greater control than the first im-

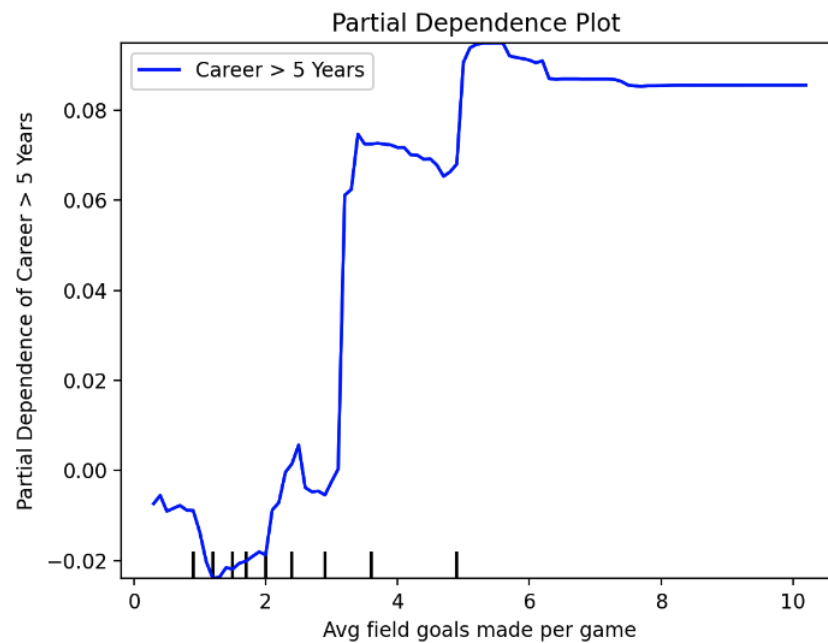


Figure 3.21: PDP NBA visual

Overall, the average marginal effect of the model predicting "Career > 5 Years" based on "Avg field goals made per game" is between -0.024866 and 0.095123.

The feature "Avg field goals made per game" directly affects the model's outcome. The partial dependence of the feature "Avg field goals made per game" on the model predicting "Career > 5 Years" starts at -0.006709.

There is a lot of variability in the output based on the feature variable. The model is most likely to predict "Career > 5 Years" based on the feature "Avg field goals made per game" when the feature is at 5 with an effect of 0.095123. The model is least likely to predict "Career > 5 Years" when the feature has a value of 1 with an effect of -0.024866.

Figure 3.22: PDP NBA text

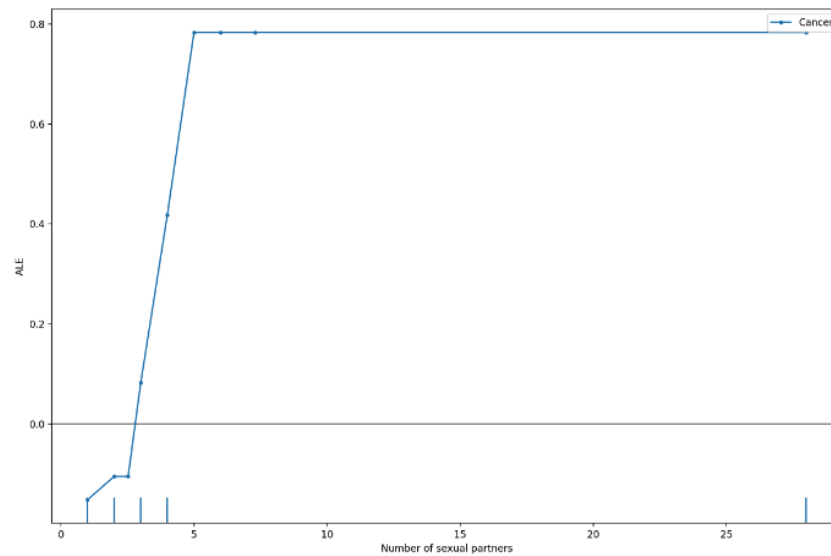


Figure 3.23: ALE cancer visual

plementation. We added a function that allowed us to check the current, previous and subsequent values in the array being looped over. We improved the logic that handled this to function more robustly. This avoided having situations where there would be "*increases, increases, increases*" and instead just have *increases* once. The rounding of numbers was improved significantly. In the first iteration it was a simple rounding that occurred by hand, whereas in the second implementation we developed a function that rounded them automatically. To overcome the rounding problem, we developed a function to decipher how many decimal places to round. We took the average number of decimal places in the inputted array and then divided it by a number set as a parameter. This figure would then be the number of decimal places to round the array. This method works well most of the time but is slightly prone to errors.

We also tested the language generation module without using the values from the LOWESS curve and just the values outputted from the partial dependence or the accumulated local effects. This required slight changes in the module's logic, but the output was the same. This begs the question of how much value the LOWESS adds to the architecture given the current architecture.

Overall, the average marginal effect of the model predicting "Cervical Cancer" based on "Number of sexual partners" is between -0.1521 and 0.7832. The feature "Number of sexual partners" directly affects the model's outcome.

The accumulated local effects of the feature "Number of sexual partners" on the model predicting "Cervical Cancer" starts with an effect of -0.1521 when the feature's value is at 1. Then, the effect increases to -0.1052 when the feature's value is at 2, stays constant at -0.1052 when the feature's value is at 2, increases to 0.7832 when the feature's value is at 5 and stays constant at 0.7832 when the feature's value is from 6 to 28.

Figure 3.24: ALE cancer text

Chapter 4

Results

4.1. Evaluation

4.1.1. Main study

To address the research question and test whether textual explanations can be more effective to users than visual explanations, we needed to conduct another study. For this study, we wanted to test which mode of explanation improved the users' overall understanding of the models and variables. This was to be an extrinsic evaluation of the language generation modules.

4.1.2. Participants

The survey was anonymous. The participants mostly fall into two categories: students of the Artificial Intelligence masters at Utrecht University or users of Reddit. Many of the students have completed their studies or are working on their thesis. The students were contacted directly or via group study and class Whatsapp groups. The survey was also posted on Reddit. We used the subreddit */r/learnmachinelearning*. We planned to use the subreddit */r/machinelearning*, but they have strict rules on users looking for participants for their studies. The learn machine learning subreddit has much less strict community guidelines. A screenshot of the post is available in the appendix.

Twenty people participated in the study. There were thirty-one participants, but eleven did not complete the survey. Some participants clicked through without answering questions and could not see which condition they experienced. Only responses that were 100% completed were used. The participants were asked: (a) how many years of experience in machine learning they had and to describe (b) their level of expertise with explainable AI (XAI) on a three-point scale: (1) Beginner, (2) Intermediate, (3) Proficient. The mean level of years of experience that the participants had was 2.55 years, with a standard deviation of 2.25. This level tells us that we had a decent mix of the target audience for this research.

4.2. Procedure

The survey was online via an online digital survey and was accessible through Qualtrics. The online element removed the interviewer effect from the experiment. Participants could complete the form later if they desired to take a break. The survey was online for a total of three weeks.

4.3. Design

The study was a between-subjects design where there were 3 (dataset) * 3(technique) design with the between-subjects manipulation for modality. This design was chosen as we decided there would be less bias if participants had not seen the other forms of explanation. There were three conditions to which users were randomly assigned: (a) visual explanations, (b) natural language explanations and (c) multimodal explanations.

A page had a brief overview of the research at the beginning and some contact information. Participants were also asked to give consent and acknowledge that

- They have reached the age of 18 years or older
- Their participation is completely voluntary
- They are aware that they can terminate the survey at any time.
- They acknowledge that anonymous responses may be used for research purposes following General Data Protection Regulation.

4.3.1. Materials

To test understanding we presented users with outputs of each mode with accompanying questions that were designed specifically to test their understanding of the outputs. For permutation

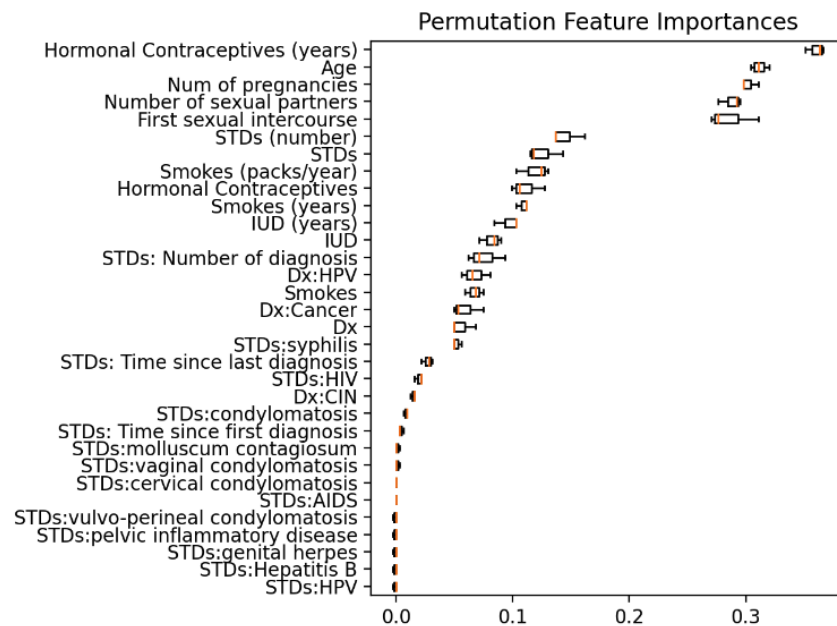


Figure 4.1: PFI cancer visual explanation

feature importance, the system was run once and output for both visual and textual explanations was taken from each one of the datasets. The visual output was kept and to be used for the visual condition, text for textual condition and a combination of both was the multimodal. The questions were designed in a way that would test user knowledge of the models' processes. Many of the questions were designed to offer likely scenarios that would happen if they were in development and looking at the outputs themselves.

For partial dependence and accumulated local effects, because they only focus on one variable and its relationship to the outcome, specific variables needed to be chosen. These variables were chosen in conjunction with the question formation as much as possible. Because the study was intended to test users' understanding of the explanations, the variables were selected when they were functioning in a way that would be contrary to intuition or that questions could be designed for. See figures 4.4, 4.5 and 4.6 for an example. In this question, the answer is contrary to what would someone might expect. In figure 4.4, it can be seen that after around 10 sexual partners, based on the model, someone who has had that much is no more likely to have cervical cancer than someone who has had 15. Therefore the answer is False.

Each participant had to answer a total of nine questions — three questions per dataset, each focusing on a different explainability technique. The order that participants experienced each dataset was randomised, but the order that the techniques were laid out was fixed. An example of the order might look something like PFI Cancer, PDP Cancer, ALE Cancer, PFI NBA, PDP NBA, ALE NBA, PFI Rain, PDP Rain and ALE Rain. Participants could not click back to previous questions as there was some similarity in the questions asked. Each dataset had a short introduction that gave a brief overview of what each dataset was about before they were presented with the questions. The questions were the same across each of the three conditions. They were designed in a way that would minimise favouring one mode of explanation over the other. All images and accompanying questions can be found in the appendix section of this paper.

4.4. Results

The main results of the study was to test users' understanding of the models and their processes. There were some interesting findings from the results of the study. First off, there were slight differences in time taken to complete the study over each condition, see figure 4.10. The text condition

Permutation feature importance has revealed that there are 25 important variables. Individually permuting these 25 variables has led to a decrease in the model's accuracy. The most important variables are "Hormonal Contraceptives (years)" and "Age". The variables "Num of pregnancies" and "Number of sexual partners" are also very important. The variables "First sexual intercourse", "STDs (number)", "STDs" and "Smokes (packs/year)" positively affect the prediction, but not a lot. Other variables shown to have a positive effect on the model's accuracy were "Hormonal Contraceptives", "Smokes (years)", "IUD (years)", "IUD", "STDs: Number of diagnosis", "Smokes", "Dx:HPV", "Dx:Cancer", "Dx", "STDs:syphilis", "STDs: Time since last diagnosis", "STDs:HIV", "Dx:CIN", "STDs:condylomatosis", "STDs: Time since first diagnosis", "STDs:molluscum contagiosum" and "STDs:vaginal condylomatosis". These variables affect the prediction, but only a tiny amount.

2 variables don't affect the model's accuracy. Permuting them individually did not change overall accuracy, either positively or negatively. The variables are "STDs:AIDS" and "STDs:cervical condylomatosis".

5 variables impact the model's accuracy negatively. Removing one of them could lead to the model's accuracy increasing. The variables are "STDs:pelvic inflammatory disease", "STDs:genital herpes", "STDs:Hepatitis B", "STDs:HPV" and "STDs:vulvo-perineal condylomatosis".

Figure 4.2: PFI cancer text

A developer wants to remove some variables as there were errors in the data collection process. Thankfully, it possible to remove some variables while maintaining a high level of accuracy.

- True
- False
- Don't know

Figure 4.3: PFI cancer question. The answer here is True. 5 variables impact the model's accuracy negatively. Removing one of them could lead to the model's accuracy increasing.

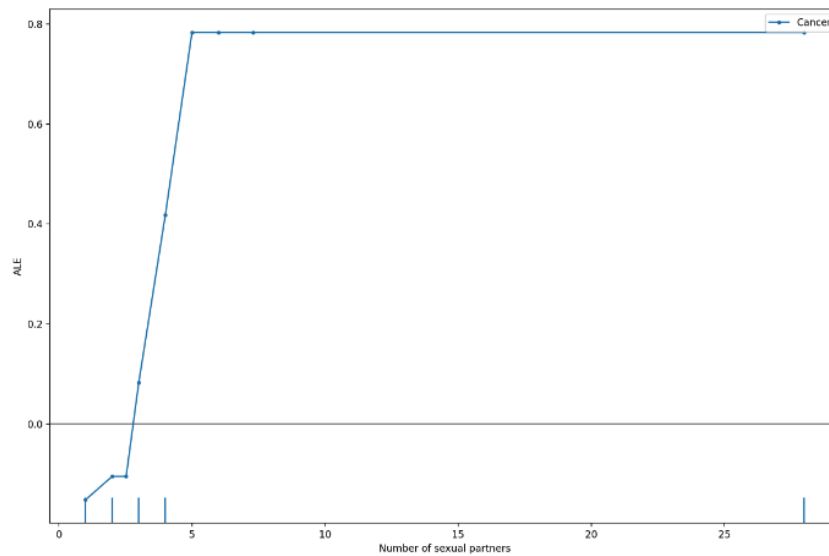


Figure 4.4: ALE cancer visual

Overall, the average marginal effect of the model predicting "Cervical Cancer" based on "Number of sexual partners" is between -0.1521 and 0.7832. The feature "Number of sexual partners" directly affects the model's outcome.

The accumulated local effects of the feature "Number of sexual partners" on the model predicting "Cervical Cancer" starts with an effect of -0.1521 when the feature's value is at 1. Then, the effect increases to -0.1052 when the feature's value is at 2, stays constant at -0.1052 when the feature's value is at 2, increases to 0.7832 when the feature's value is at 5 and stays constant at 0.7832 when the feature's value is from 6 to 28.

Figure 4.5: ALE cancer text

Sandra has had 10 sexual partners. Based on the model, she is more at risk to have cervical cancer than her friend who has had 15.

- True
- False
- Don't know

Figure 4.6: ALE cancer question

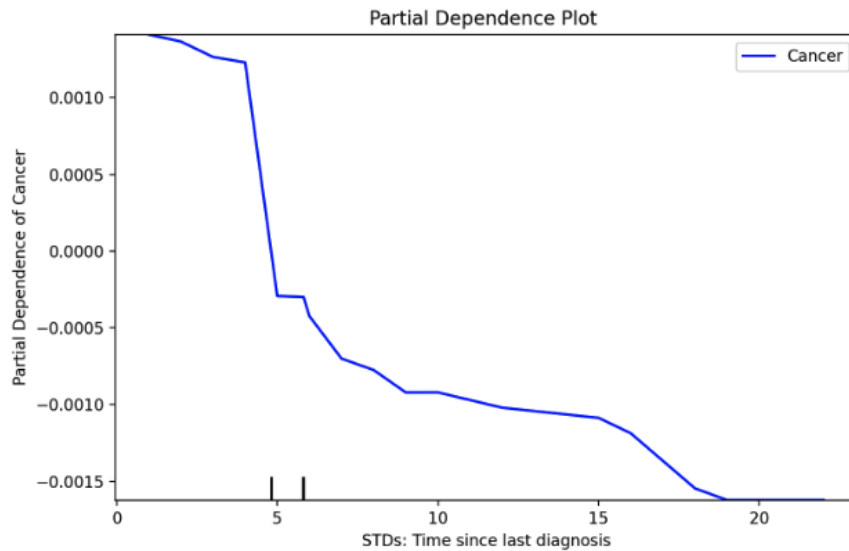


Figure 4.7: PDP cancer visual

Overall, the average marginal effect of the model predicting "Cervical Cancer" based on "STDs: Time since last diagnosis" is between -0.00147 and 0.002218. The feature "STDs: Time since last diagnosis" directly affects the model's outcome. The partial dependence of the feature "STDs: Time since last diagnosis" on the model predicting "Cervical Cancer" starts at 0.002218.

There is a lot of variability in the output based on the feature variable. The model is most likely to predict "Cervical Cancer" based on the feature "STDs: Time since last diagnosis" when the feature is at 1 with an effect of 0.002218. The model is least likely to predict "Cervical Cancer" when the feature has a value of 19 with an effect of -0.00147.

Figure 4.8: PDP cancer text

Based on the model, there is no relationship between "STDs: Time since last diagnosis" and cervical cancer.

- True
- False
- Don't know

Figure 4.9: PDP cancer question

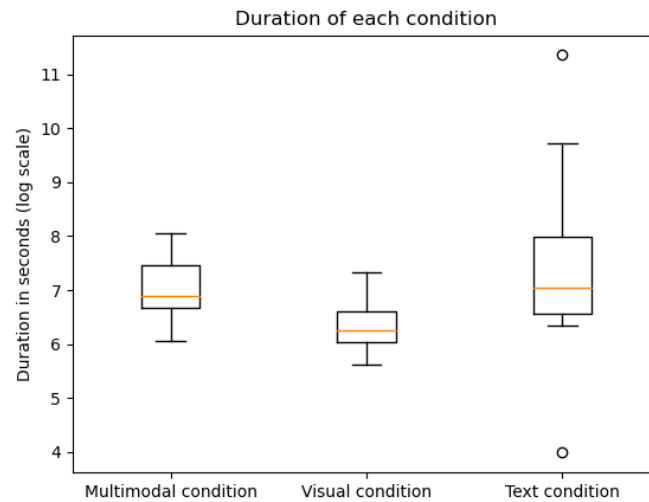


Figure 4.10: Duration of each condition log scale

Condition	Multimodal	Visual	Text
Num. Participants	7	8	8

Table 4.1: Number of participants per condition

took the longest to complete, followed by the multimodal condition. This tells us that the text might have slowed the process down somewhat.

4.4.1. Removal of outliers based on the log-scale

The multimodal condition scored the best out of the three conditions, followed by the visual condition then the text condition came last. As can be seen from figure 4.11, there are no outliers in the graph, but there is a long tail on the multimodal condition. However, when plotting on a log scale (see figure 4.12, there is one outlier in the multimodal case. We decided that since there was a small number of participants in the study, that the feedback was more sensitive to outliers. When performing the log-transform, we reduce the variance but points at the edge of the distribution became more distinct as being further from the rest of the distribution. We decided to remove these points. This resulted in the removal of three participants' results.

4.4.2. Results

As can be seen from figure 4.14, there is a clear difference between all three conditions. The multimodal condition clearly does the best based on the mean score. The multimodal is then followed by the visual condition and the text condition did the worst. It should also be noted that there is a much wider spread of correct answers in the text condition than the other two.

4.4.3. ANCOVA

To test whether these differences were significant, we decided to use a one-way ANCOVA. ANCOVA is a general linear model which is a combination of ANOVA and regression. ANCOVA evaluates whether the means of a dependent variable are equal across levels of a categorical variable, while controlling for the effects of other variables that are not of primary interest, known as covariates. In this experiment, the dependent variable was the percentage of correct answers and the independent variable was group condition assignment. Before running any significance tests, the assumptions were first checked. These included checking:

- **That the co-variate was continuous**

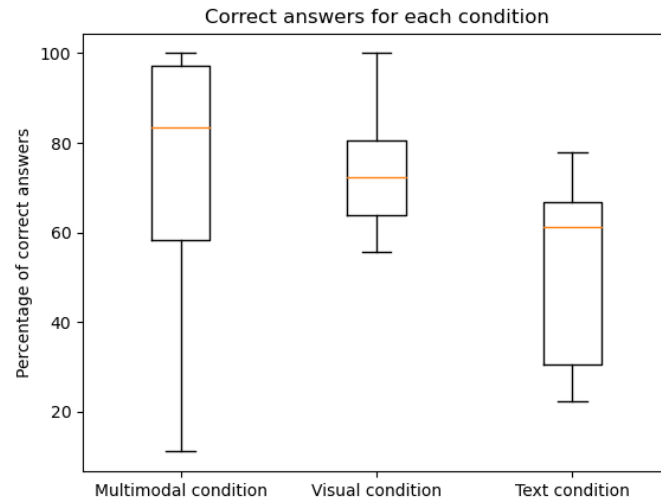


Figure 4.11: Percentage of correct answers for each condition (outliers included)

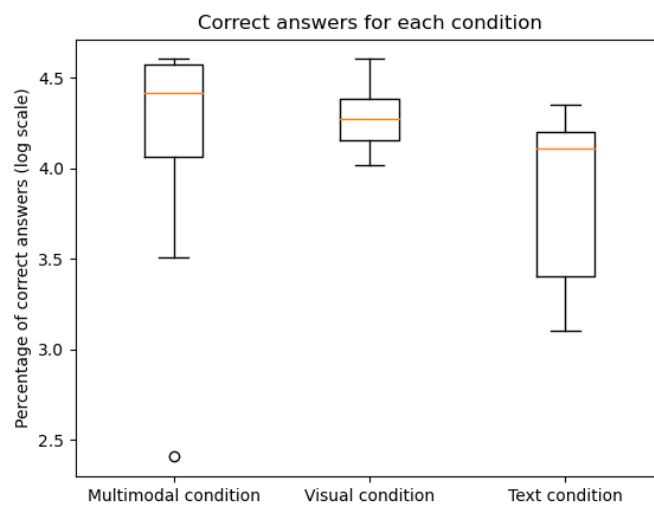


Figure 4.12: Percentage of correct answers for each condition log transformed (outliers included)

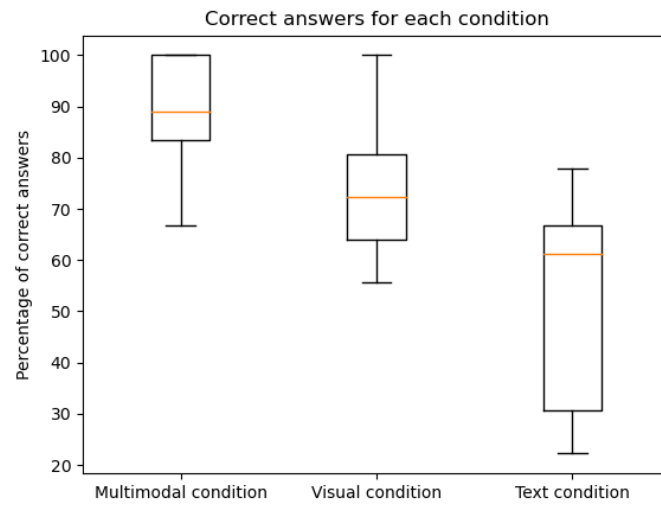


Figure 4.13: Percentage of correct answers for each condition (outliers removed)

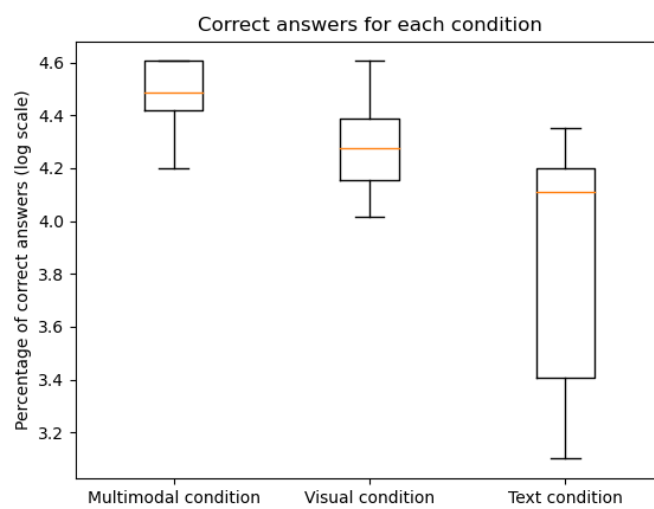


Figure 4.14: Percentage of correct answers for each condition log (outliers removed)

Source	Sum of Squares	Degrees of Freedom	F-values	P-values	Partial eta squared
Mode	225	2	5.336670	0.018937	0.432586
Residual	3742	14	NaN	NaN	NaN

Table 4.2: ANCOVA results

Group 1	Group 2	Meandiff	P-Adj	Lower	Upper	Reject
Multimodal	Text	-31.1111	0.0136	-55.8453	-6.377	True
Multimodal	Visual	-14.8148	0.2611	-38.3159	8.6862	False
Text	Visual	16.2963	0.2544	-9.2823	41.8749	False

Table 4.3: Multiple comparison of means - Turkey HSD, FWER = 0.05

- **There were no extreme outliers:** Although defining outliers based off of the log scale is a less common technique, we feel like it is justified given the small number of participants.
- **Data was normally distributed:** To test whether data was normally distributed, we used the Shapiro-Wilk test. Results of the Shapiro-Wilk test on each condition were the following: multimodal ($\chi^2 = 0.86$, $p = 0.14$), visual ($\chi^2 = 0.93$, $p = 0.56$) and for the text condition ($\chi^2 = 0.85$, $p = 0.1$). This tells us that the data from all three conditions is normally distributed.
- **The co-variate and the independent variable were independent of one another:** The group assignment to explainability technique was independent from years of experience.
- **Homogeneity of variances among groups are roughly equal:** Bartlett's test for homogeneity of variances was used to test for homogeneity. The result showed that variances were largely homogeneous ($\chi^2 = 1.91$, $p = 0.38$).
- **Observations in each group were independent**

The null hypothesis says that there should be no difference in the number of correct responses across conditions. With a p-value of 0.02 we can therefore reject null hypothesis that all explanations are equally useful. The full results of the ANCOVA can be found in table 4.4.3., Although the ANCOVA tells us that there is a significant difference between the groups, it doesn't tell us exactly what that difference is. For this we need a Tukey's HSD test which is a multiple comparison statistical test that compares all possible pairs. Below are the results of the Tukey HSD As can be seen from table 4.3, there is a significant difference between the multimodal and text condition ($p = 0.01$). This directly answers the second research question; that multimodal explanations are significantly better than individual modes of explanation. It does not, however, tell us that multimodal explanations are better than *all* modes of explanation. Namely, the multimodal condition did not do significantly better than the standard visual format of explanation.

Chapter 5

Discussion

This work has addressed the following questions in terms of user understanding of explanations in the AutoML context:

- Are textual explanations more effective than visual explanations?
- Are visual and textual explanations combined more effective than individual modes?

To address these research questions, template-based generation modules were built to produce natural language explanations for permutation feature importance, partial dependence and accumulated local effects. We tested these modules in a pilot study to gain more knowledge about what needed to be changed. To interpret the raw data outputs from these explainability techniques, local regression and clustering steps were employed. Following this, the main research study was to test participant's understanding of the information that the visual, textual and multimodal conditions produced. The survey questions were carefully designed to test user's understanding. To test for significance, results were ran through a ANCOVA which found a significant relationship between groups. Following this, a Tukey's HSD was run to find where the significant difference was.

This research has found that there is no significant difference in the effectiveness between textual explanations and visual explanations. Therefore the main research question has not been answered. However, the secondary research question has been addressed, at least partially so. Multimodal explanations are more effective than individual modes. But multimodal explanations are not better than *all* individual modes. Unfortunately it is hard to draw any solid conclusions from such a small number of study participants, therefore these findings are preliminary and call for followup research. Although it was not enough to be significant, the multimodal condition did do slightly better than the visual.

There is always a chance of bias in the survey questions toward individual conditions or techniques. The survey questions could be a potential limitation but challenging to minimise in a between-subject comprehension questionnaire.

An interesting finding from the study was that the duration differed between groups 4.10. This finding could mean multiple things. The most interesting is that the participants assigned to the visual condition were the quickest to finish the survey but did not score as highly for the number of correct answers as the multimodal. It could be possible that participants in the visual condition thought that the answers were more evident than those in the multimodal condition. The text condition had the widest spread of data. Because this was the condition where participants scored the lowest, some people could have taken a long time because questions were difficult or skipped through quickly because they were not immediately apparent.

5.0.1. Local methods

Although this project is focused on global methods, local methods are another direction that could be taken to improve autoML. Counterfactual explanations would offer what-if scenarios if individual values that led to a prediction differed. Offering users contrastive methods seems like a compelling way to explain to people how a prediction was made [26]. Counterfactual methods find their roots in philosophy and grounding how humans explain things in real life. Using contrastive explanations could be a good way of explaining through natural language. Local model-agnostic methods such as LIME [35], SHAP and Shapley are widely used [25]. These local methods have been used for AutoML systems elsewhere [39].

A limitation of the study was that there were not enough participants, so it is difficult to draw concrete conclusions from it. Given more time, this research would be scaled up to increase n to allow for an effective conclusion. We took quite an aggressive approach to remove outliers based on the log-transformed values. We decided that this was justified to avoid susceptibility to unwanted variance since there was such a small number of participants in the study. We would not have taken such an aggressive approach to outlier removal if more participants were in the survey. When the ANCOVA was run with outliers left in, there were no significant differences between the groups.

We felt that the survey was already quite long during the survey design, so these steps seemed unnecessary. No data was taken on gender differences or nationality. If the survey were conducted again, it would be interesting to see how gender influences which mode leads to better utility, such as what was found in Gkatzia et al.'s paper [13]. As well as this, cultural differences and the level of English might play a significant role. These demographics could have been a factor contributing to the study's results. For example, perhaps there were more people with English as their second language in the text condition, which might have skewed the results to favour the visual explanation condition.

Another approach we could have taken in the evaluation was a within-subjects design to get higher effects from the small number of participants. However, we decided early on that a more significant bias could have been associated with a within-subjects design.

5.0.1.1. *Clustering*

The clustering step in permutation feature importance is a clear area that could be improved. If more time allowed, further considerations would be spent on finding a better way of clustering the important variables. The clustering works well but sometimes does not when applied to different features or different data sets. Even though three different clustering algorithms were experimented with extensively (DBSCAN, K-Means, Affinity Propagation), further exploration of alternate algorithms would greatly help the language generation module in this area. Part of choosing K-means was that it allowed for a fixed number of outputs. This fixed number allowed us to design language better.

As outlined before, the rounding of numbers is not as dynamic to variable inputs as it could be, even after the second implementation. The number of decimal places to be rounded to is calculated as follows; $(\text{average number of decimal places}) / (\text{rounding parameter})$. This number is the number of decimal places the array would be rounded. This strategy worked quite well for most of the inputs but sometimes did not work well, and the threshold parameter needed to be adjusted. Future work would find a better way of performing this rounding.

There is no doubt that natural language can, at the very least, provide an extra layer of explanation to the visual formats. Future work might include alternate explainability techniques or focus on an entirely new set of explainability techniques. As this work only focused on three global model-agnostic techniques, a likely avenue could continue and improve on this work or focus on only local model-agnostic techniques. An easy extension to this project would be to load LOCO importance into the permutation feature importance generation module. Another extension could be to extend the PDP and ALE module to produce explanations for ICE plots. This would involve considerably more work than the PFI/LOCO extension and perhaps only work well if datasets with very small numbers of instances are used.

One part of this research that took a significant amount of implementation time that, in a late stage, we needed to minimise to avoid repetition was the descriptions of each "increase, decrease..." in the line for partial dependence and accumulated local effects. The other sections of the ALE/PDP language module were relatively straightforward to implement, but when we increased the number of fits after the preliminary pilot study to increase fidelity in values, the more repetitive the language got. We attempted to find a balance between having a low number of fits where only macro changes would be described while maintaining a high level of fidelity when describing the points that these changes happened, but this seemed very difficult. We decided to set a threshold for how many times the "increases..., decreases..., increases..., " repeated. We concluded that repeating the sequence over five times could be too repetitive and sound robotic. An alternate method of using the local regression to find macro changes might be used in future while using raw data values to describe the maximum and minimum values of where these increases happen might be helpful, but mixing these two might be a challenge. The language generation modules focus purely on classification problems, and the PDP and ALE only focus on numerical variables. These are likely avenues to expand the modules further in the future.

Chapter 6

Conclusion

This research has provided insight into natural language explanations and will act as a starting point for the area. Although there were not enough participants to draw clear conclusions, multi-modal explanations have been shown to outperform visual explanations. Given that the text-only condition did the worst out of the three explanation modes, it remains to be seen whether the text can be improved further. Since the area of natural language explanations is unexplored and is only something that has been talked about, this work can provide groundwork on which further research can be conducted.

Improvements could focus upon the clustering step of permutation feature importance. This step could be prone to errors when changing datasets. Another area that has the potential to see huge improvements is the rounding of numbers, as well as perhaps using a combination of LOWESS and raw outputs from PD/ALE rather than using one or the other. We focused purely on numerical variables to narrow the focus for partial dependence and accumulated local effects. More research could be done to extend the generation module to include regression models or categorical variables.

Incomplete or poorly understood models are still barriers to entry for machine learning practitioners. With better computing power, algorithms, and low-code to no-code systems, better explanations are becoming more of a necessity than a requirement. Better explanations can help us peer into the box within the autoML space so that models can be deployed with trust and regularity.

Bibliography

- [1] Amina Adadi and Mohammed Berrada. "Peeking inside the black-box: a survey on explainable artificial intelligence (XAI)". In: *IEEE access* 6 (2018), pp. 52138–52160.
- [2] Behnaz Arzani, Kevin Hsieh, and Haoxian Chen. "Interpret-able feedback for AutoML systems". In: *arXiv preprint arXiv:2102.11267* (2021).
- [3] Maria Cheung and Myra Fitzpatrick. "The impact of the CervicalCheck controversy on provision of colposcopy services in Ireland: a cohort study". In: *European Journal of Obstetrics & Gynecology and Reproductive Biology* (2021).
- [4] Miruna-Adriana Clinciu and Helen Hastie. "A survey of explainable AI terminology". In: *Proceedings of the 1st Workshop on Interactive Natural Language Technology for Explainable Artificial Intelligence (NL4XAI 2019)*. 2019, pp. 8–13.
- [5] Ashley Deeks. "The judicial demand for explainable artificial intelligence". In: *Columbia Law Review* 119.7 (2019), pp. 1829–1850.
- [6] Kees Van Deemter, Mariët Theune, and Emiel Kraemer. "Real versus template-based natural language generation: A false opposition?". In: *Computational linguistics* 31.1 (2005), pp. 15–24.
- [7] Finale Doshi-Velez and Been Kim. "Towards a rigorous science of interpretable machine learning". In: *arXiv preprint arXiv:1702.08608* (2017).
- [8] Filip Karlo Došilović, Mario Brčić, and Nikica Hlupić. "Explainable artificial intelligence: A survey". In: *2018 41st International convention on information and communication technology, electronics and microelectronics (MIPRO)*. IEEE. 2018, pp. 0210–0215.
- [9] Amitai Etzioni and Oren Etzioni. "Designing AI systems that obey our laws and values". In: *Communications of the ACM* 59.9 (2016), pp. 29–31.
- [10] Kelwin Fernandes, Jaime S Cardoso, and Jessica Fernandes. "Transfer learning with partial observability applied to cervical cancer screening". In: *Iberian conference on pattern recognition and image analysis*. Springer. 2017, pp. 243–250.
- [11] Albert Gatt and Emiel Kraemer. "Survey of the state of the art in natural language generation: Core tasks, applications and evaluation". In: *Journal of Artificial Intelligence Research* 61 (2018), pp. 65–170.
- [12] Pieter Gijsbers et al. "An open source AutoML benchmark". In: *arXiv preprint arXiv:1907.00909* (2019).
- [13] Dimitra Gkatzia, Oliver Lemon, and Verena Rieser. "Natural language generation enhances human decision-making with uncertain information". In: *arXiv preprint arXiv:1606.03254* (2016).
- [14] Bryce Goodman and Seth Flaxman. "European Union regulations on algorithmic decision-making and a "right to explanation"". In: *AI magazine* 38.3 (2017), pp. 50–57.
- [15] Alicja Gosiewska, Anna Kozak, and Przemysław Biecek. "Simpler is better: Lifting interpretability-performance trade-off via automated feature engineering". In: *Decision Support Systems* (2021), p. 113556.
- [16] Xin He, Kaiyong Zhao, and Xiaowen Chu. "AutoML: A Survey of the State-of-the-Art". In: *Knowledge-Based Systems* 212 (2021), p. 106622.
- [17] Andreas Holzinger et al. "What do we need to build explainable AI systems for the medical domain?" In: *arXiv preprint arXiv:1712.09923* (2017).
- [18] David M Howcroft et al. "Twenty years of confusion in human evaluation: NLG needs evaluation sheets and standardised definitions". In: *Proceedings of the 13th International Conference on Natural Language Generation*. 2020, pp. 169–182.
- [19] Harmanpreet Kaur et al. "Interpreting Interpretability: Understanding Data Scientists' Use of Interpretability Tools for Machine Learning". In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 2020, pp. 1–14.

- [20] Alexandra Kirsch. "Explain to whom? Putting the User in the Center of Explainable AI". In: *Proceedings of the First International Workshop on Comprehensibility and Explanation in AI and ML 2017 co-located with 16th International Conference of the Italian Association for Artificial Intelligence (AI* IA 2017)*. 2017.
- [21] Anna S Law et al. "A comparison of graphical and textual presentations of time series data to support medical decision making in the neonatal intensive care unit". In: *Journal of clinical monitoring and computing* 19.3 (2005), pp. 183–194.
- [22] Xiaochi Liu et al. "Novel application of machine learning algorithms and model-agnostic methods to identify factors influencing childhood blood lead levels". In: *Environmental science & technology* 55.19 (2021), pp. 13387–13399.
- [23] Samuele Lo Piano. "Ethical principles in machine learning and artificial intelligence: cases from the field and possible ways forward". In: *Humanities and Social Sciences Communications* 7.1 (2020), pp. 1–7.
- [24] Bureau Of Meteorology. *Climate data online*. 2015.
- [25] Christoph Molnar. *Interpretable machine learning*. Lulu. com, 2020.
- [26] Christoph Molnar, Giuseppe Casalicchio, and Bernd Bischl. "Interpretable machine learning—a brief history, state-of-the-art and challenges". In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer. 2020, pp. 417–431.
- [27] Christoph Molnar et al. "General pitfalls of model-agnostic interpretation methods for machine learning models". In: *arXiv preprint arXiv:2007.04131* (2020).
- [28] Shweta Narkar et al. "Model LineUpper: Supporting Interactive Model Comparison at Multiple Levels for AutoML". In: *26th International Conference on Intelligent User Interfaces*. 2021, pp. 170–174.
- [29] Fearghal O'Donncha et al. "Data driven insight into fish behaviour and their use for precision aquaculture". In: *Frontiers in Animal Science* (2021), p. 30.
- [30] Dong Huk Park et al. "Multimodal explanations: Justifying decisions and pointing to the evidence". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 8779–8788.
- [31] François Portet et al. "Automatic generation of textual summaries from neonatal intensive care data". In: *Conference on Artificial Intelligence in Medicine in Europe*. Springer. 2007, pp. 227–236.
- [32] Esteban Real et al. "Automl-zero: Evolving machine learning algorithms from scratch". In: *International Conference on Machine Learning*. PMLR. 2020, pp. 8007–8019.
- [33] Ehud Reiter. "Natural language generation challenges for explainable AI". In: *arXiv preprint arXiv:1911.08794* (2019).
- [34] Ehud Reiter and Robert Dale. "Building applied natural language generation systems". In: *Natural Language Engineering* 3.1 (1997), pp. 57–87.
- [35] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "" Why should i trust you?" Explaining the predictions of any classifier". In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 2016, pp. 1135–1144.
- [36] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Model-agnostic interpretability of machine learning". In: *arXiv preprint arXiv:1606.05386* (2016).
- [37] Cynthia Rudin. "Please stop explaining black box models for high stakes decisions". In: *stat* 1050 (2018), p. 26.
- [38] Advait Siddharthan et al. "Blogging birds: Generating narratives about reintroduced species to promote public engagement". In: *INLG 2012 Proceedings of the Seventh International Natural Language Generation Conference*. 2012, pp. 120–124.

- [39] Anh Truong et al. "Towards automated machine learning: Evaluation and comparison of AutoML approaches and tools". In: *2019 IEEE 31st international conference on tools with artificial intelligence (ICTAI)*. IEEE. 2019, pp. 1471–1479.
- [40] Marian Van Der Meulen et al. "When a graph is poorer than 100 words: A comparison of computerised natural language generation, human generated descriptions and graphical displays in neonatal intensive care". In: *Applied Cognitive Psychology: The Official Journal of the Society for Applied Research in Memory and Cognition* 24.1 (2010), pp. 77–89.
- [41] Bochao Wang et al. "Vega: towards an end-to-end configurable automl pipeline". In: *arXiv preprint arXiv:2011.01507* (2020).
- [42] Iordanis Xanthopoulos et al. "Putting the Human Back in the AutoML Loop." In: *EDBT/ICDT Workshops*. 2020.
- [43] Y Yu et al. "An approach to generating summaries of time series data in the gas turbine domain". In: *2001 International Conferences on Info-Tech and Info-Net. Proceedings (Cat. No. 01EX479)*. Vol. 3. IEEE. 2001, pp. 44–51.
- [44] Syed Zafar. *ML classification: Career Longevity for NBA players*. Sept. 2017. URL: <https://data.world/ssaudz/ml-classification-predicting-5-year-career-longevity-for-nb>.
- [45] Barret Zoph and Quoc V Le. "Neural architecture search with reinforcement learning". In: *arXiv preprint arXiv:1611.01578* (2016).

Appendix A

Appendix

All code used for this project is freely available at <https://github.com/davidpaulniland/Natural-Language-Explanations.git>

A.1. Main study

A.1.1. Permutation feature importance

A.1.1.1. Cervical cancer dataset

A.1.1.2. NBA dataset

A.1.1.3. Rain dataset

A.1.2. Partial dependence

A.1.2.1. Cervical cancer dataset

A.1.2.2. NBA dataset

A.1.2.3. Rain dataset

A.1.3. Accumulated local effects

A.1.3.1. Cervical cancer dataset

A.1.3.2. NBA dataset

A.1.3.3. Rain dataset

 /r/learnmachinelearning · Posted by  13 days ago 

 1  **I'm looking for participants for my thesis research which aims to explain black-box algorithms better.**

Request

The survey is short.

It isn't essential that you have much background in machine learning or explainability.

I really appreciate anyone who takes the time to go through the survey.

https://survey.uu.nl/jfe/form/SV_6LGWCp0C3zmidnM

Thank you for reading!

 0 Comments  Share  Edit Post  Save  Hide   Tip ...

Post Insights
Only you and mods of this community can see this

1.5k 	100%	2	0
 Total Views	 Upvote Rate	 Community Karma	 Total Shares



Figure A.1: Post on Reddit.com/r/LearnMachineLearning

Natural Language Explanations

This research is focused on testing whether users find textual explanations for machine learning models more useful than visual explanations. Although some people might prefer information communicated visually, this might not necessarily lead to better decision making and analysis.

This survey is a preliminary study to test whether generated text reflects what is in the graphs they are based on. The explainability techniques interrogate how models use variables in making their predictions.

The first example is of an algorithm that predicts cervical cancer. The second example is an algorithm that predicts whether it will rain tomorrow in Australia. The study is aimed at making machine learning more intuitive and user friendly. Therefore, it isn't essential that you know a lot about machine learning or explainable AI (XAI).

 davidpaulniland@gmail.com (not shared) [Switch account](#) 

*** Required**

About this survey

This survey is part of a master's thesis in artificial intelligence at Utrecht University. It is conducted by David-Paul Niland. Should you have any questions please get in contact (d.p.niland@students.uu.nl).

Informed consent

By clicking agree to the terms and conditions, you acknowledge that

- You have reached the age of 18 years or older
- Your participation is completely voluntary.
- You are aware that you can terminate the survey at any time.
- You acknowledge that your anonymous responses may be used for research purposes in accordance with General Data Protection Regulation.

Figure A.2: Front page preliminary pilot study

Informed Consent *

I understand the terms and conditions of this survey.

[Next](#) Page 1 of 8 [Clear form](#)

Never submit passwords through Google Forms.

Figure A.3: Consent preliminary pilot study

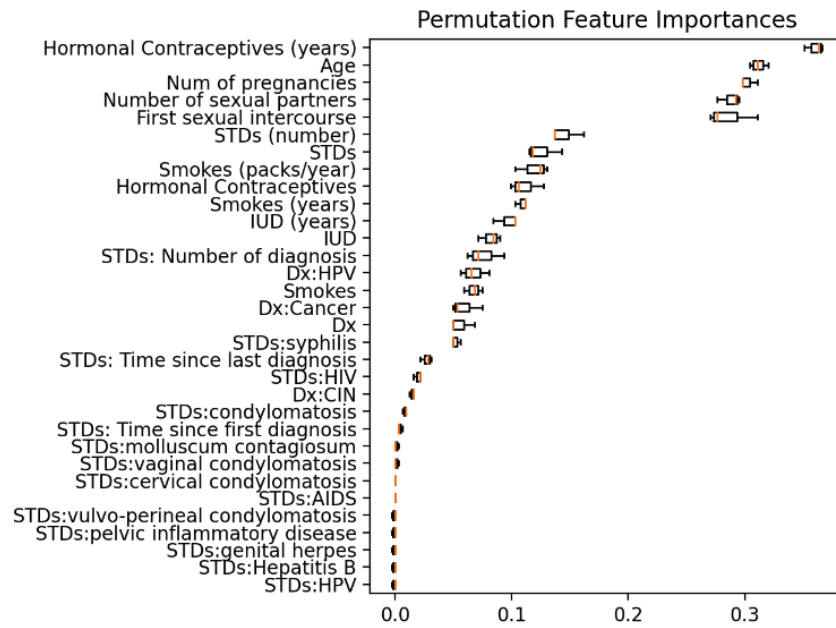


Figure A.4: PFI cancer visual explanation

Permutation feature importance has revealed that there are 25 important variables. Individually permuting these 25 variables has led to a decrease in the model's accuracy. The most important variables are "Hormonal Contraceptives (years)" and "Age". The variables "Num of pregnancies" and "Number of sexual partners" are also very important. The variables "First sexual intercourse", "STDs (number)", "STDs" and "Smokes (packs/year)" positively affect the prediction, but not a lot. Other variables shown to have a positive effect on the model's accuracy were "Hormonal Contraceptives", "Smokes (years)", "IUD (years)", "IUD", "STDs: Number of diagnosis", "Smokes", "Dx:HPV", "Dx:Cancer", "Dx", "STDs:syphilis", "STDs: Time since last diagnosis", "STDs:HIV", "Dx:CIN", "STDs:condylomatosis", "STDs: Time since first diagnosis", "STDs:molluscum contagiosum" and "STDs:vaginal condylomatosis". These variables affect the prediction, but only a tiny amount.

2 variables don't affect the model's accuracy. Permuting them individually did not change overall accuracy, either positively or negatively. The variables are "STDs:AIDS" and "STDs:cervical condylomatosis".

5 variables impact the model's accuracy negatively. Removing one of them could lead to the model's accuracy increasing. The variables are "STDs:pelvic inflammatory disease", "STDs:genital herpes", "STDs:Hepatitis B", "STDs:HPV" and "STDs:vulvo-perineal condylomatosis".

Figure A.5: PFI cancer text

A developer wants to remove some variables as there were errors in the data collection process. Thankfully, it possible to remove some variables while maintaining a high level of accuracy.

- True
- False
- Don't know

Figure A.6: PFI cancer question

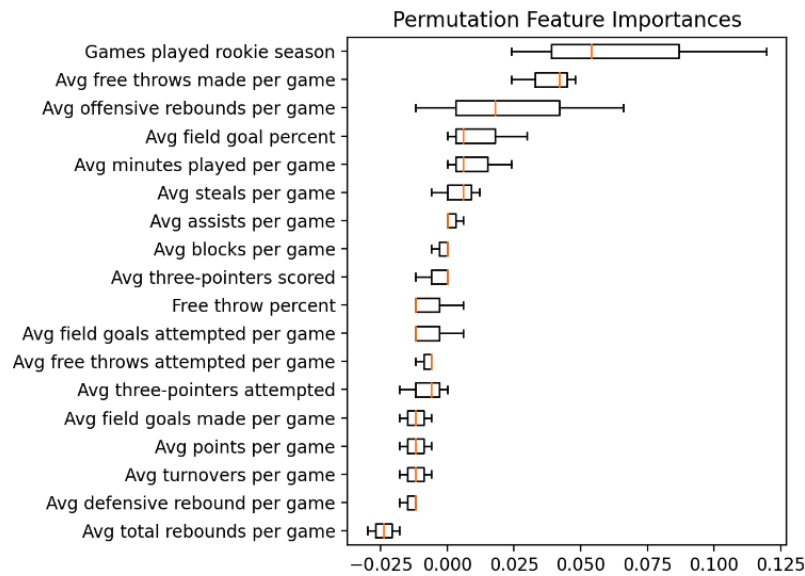


Figure A.7: PFI NBA visual

Permutation feature importance has revealed that there are 7 important variables. Individually permuting these 7 variables has led to a decrease in the model's accuracy. The most important variable is "Games played rookie season". The variable "Avg free throws made per game" is also very important. The variable "Avg offensive rebounds per game" positively affects the prediction, but not a lot. The variable "Avg field goal percent" was shown to have a positive effect on the model's accuracy, but only a very small amount. "Avg minutes played per game", "Avg steals per game" and "Avg assists per game". These variables affect the prediction, but only a tiny amount.

11 variables impact the model's accuracy negatively. Removing one of them could lead to the model's accuracy increasing. The variables are "Avg blocks per game", "Avg three-pointers scored", "Avg field goals attempted per game", "Free throw percent", "Avg free throws attempted per game", "Avg three-pointers attempted", "Avg field goals made per game", "Avg points per game", "Avg turnovers per game", "Avg defensive rebound per game" and "Avg total rebounds per game".

Figure A.8: PFI NBA text

A developer would like to make a more simple model. They consider removing "Avg free throws per game". Is this a good variable to remove?

Yes

No

Don't know

Figure A.9: PFI NBA question

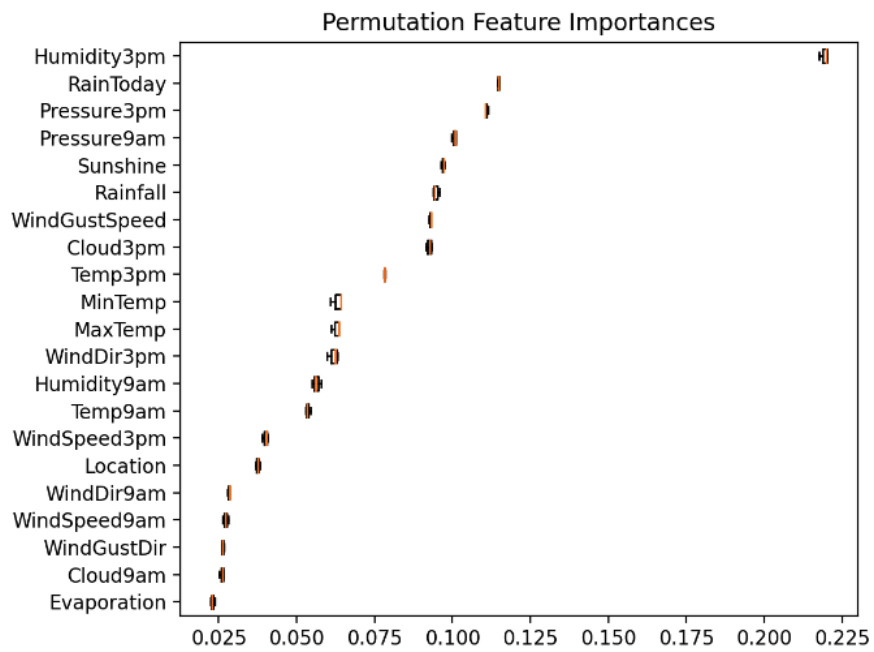


Figure A.10:

Permutation feature importance has revealed that all 21 variables are important. Individually permuting these variables has led to a decrease in the model's accuracy. The most important variables are "Humidity3pm", "Pressure3pm" and "RainToday". The variables "Sunshine", "WindGustSpeed" and "Pressure9am" are very important as well. The variables "Temp3pm", "Cloud3pm" and "Rainfall" have some positive effects on the prediction, but not a lot. Other variables that were shown to have a positive effect on the model's accuracy were "Humidity9am", "MinTemp", "Temp9am", "WindDir3pm", "MaxTemp", "WindDir9am", "WindSpeed3pm", "Location", "WindSpeed9am", "Cloud9am", "WindGustDir" and "Evaporation". These variables have some effect on the prediction, but only a very small amount.

Figure A.11: PFI rain text

Humidity3pm is the only variable contributing towards the prediction.

True

False

Don't know

Figure A.12: PFI rain question

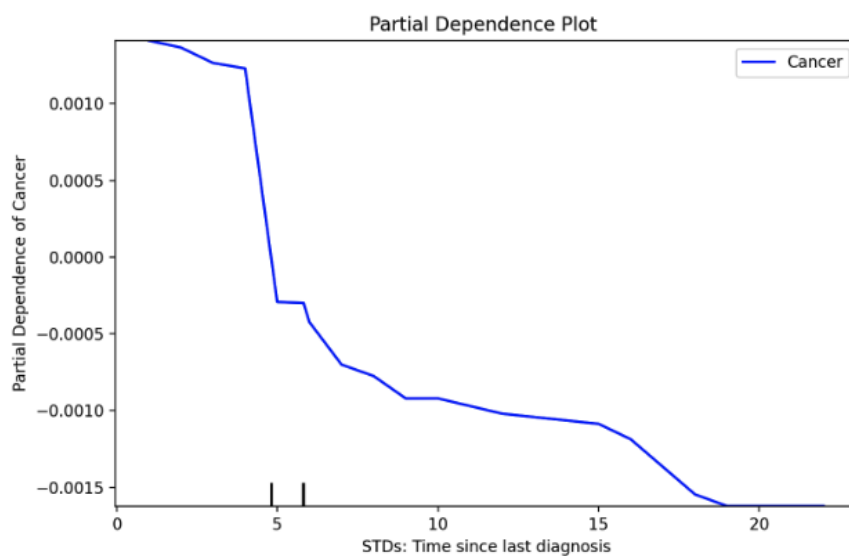


Figure A.13: PDP cancer visual

Overall, the average marginal effect of the model predicting "Cervical Cancer" based on "STDs: Time since last diagnosis" is between -0.00147 and 0.002218. The feature "STDs: Time since last diagnosis" directly affects the model's outcome. The partial dependence of the feature "STDs: Time since last diagnosis" on the model predicting "Cervical Cancer" starts at 0.002218.

There is a lot of variability in the output based on the feature variable. The model is most likely to predict "Cervical Cancer" based on the feature "STDs: Time since last diagnosis" when the feature is at 1 with an effect of 0.002218. The model is least likely to predict "Cervical Cancer" when the feature has a value of 19 with an effect of -0.00147.

Figure A.14: PDP cancer text

Based on the model, there is no relationship between "STDs: Time since last diagnosis" and cervical cancer.

- True
- False
- Don't know

Figure A.15: PDP cancer question

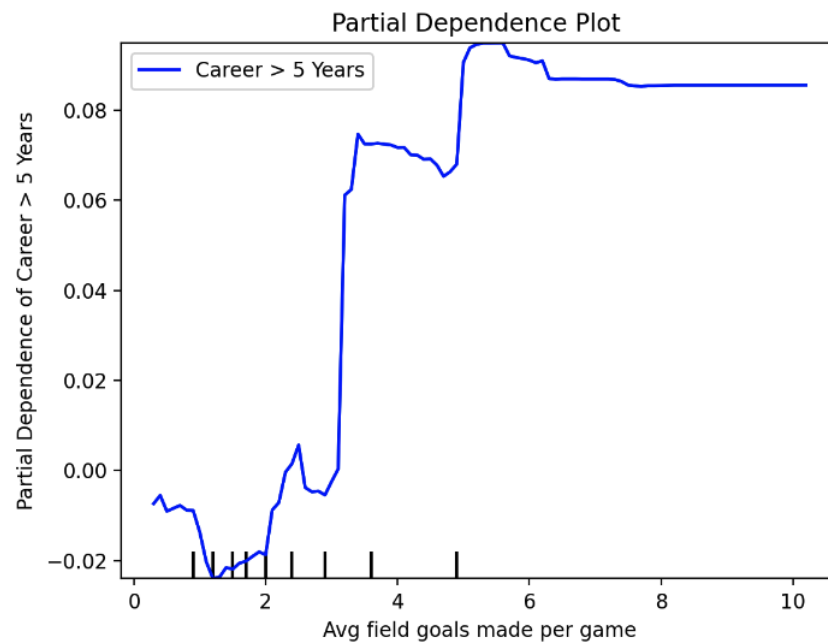


Figure A.16: PDP NBA visual

Overall, the average marginal effect of the model predicting "Career > 5 Years" based on "Avg field goals made per game" is between -0.024866 and 0.095123.

The feature "Avg field goals made per game" directly affects the model's outcome. The partial dependence of the feature "Avg field goals made per game" on the model predicting "Career > 5 Years" starts at -0.006709.

There is a lot of variability in the output based on the feature variable. The model is most likely to predict "Career > 5 Years" based on the feature "Avg field goals made per game" when the feature is at 5 with an effect of 0.095123. The model is least likely to predict "Career > 5 Years" when the feature has a value of 1 with an effect of -0.024866.

Figure A.17: PDP NBA text

A player is scoring one field goal per game. He's likely to stay on the team for more than five years.

- True
- False
- Don't know

Figure A.18: PDP NBA question

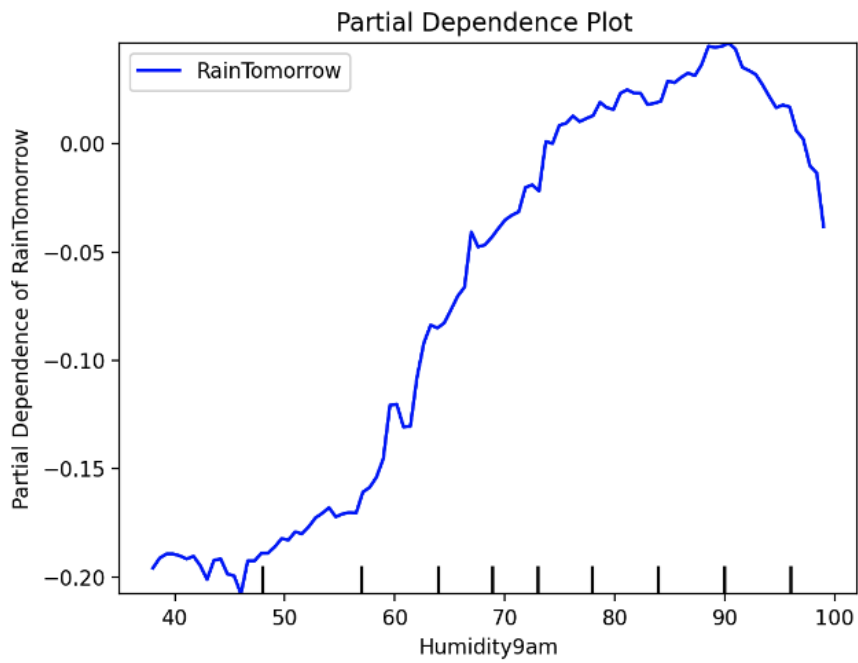


Figure A.19: PDP rain visual

Overall, the average marginal effect of the model predicting "RainTomorrow" based on "Humidity9am" is between -0.201 and 0.046. The feature "Humidity9am" directly affects the model's outcome. The partial dependence of the feature "Humidity9am" on the model predicting "RainTomorrow" starts at -0.194.

There is a lot of variability in the output based on the feature variable. The model is most likely to predict "RainTomorrow" based on the feature "Humidity9am" when the feature is at 90 with an effect of 0.046. The model is least likely to predict "RainTomorrow" when the feature has a value of 46 with an effect of -0.201.

Figure A.20: PDP rain text

Humidity is at 46 in the morning. Based on the model, it will likely rain tomorrow.

True

False

Don't know

Figure A.21: PDP rain question

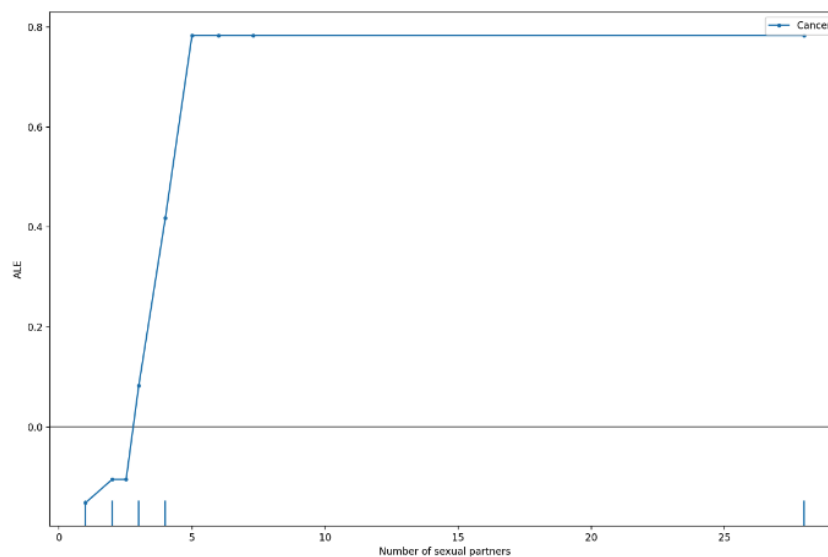


Figure A.22: ALE cancer visual

Overall, the average marginal effect of the model predicting "Cervical Cancer" based on "Number of sexual partners" is between -0.1521 and 0.7832. The feature "Number of sexual partners" directly affects the model's outcome.

The accumulated local effects of the feature "Number of sexual partners" on the model predicting "Cervical Cancer" starts with an effect of -0.1521 when the feature's value is at 1. Then, the effect increases to -0.1052 when the feature's value is at 2, stays constant at -0.1052 when the feature's value is at 2, increases to 0.7832 when the feature's value is at 5 and stays constant at 0.7832 when the feature's value is from 6 to 28.

Figure A.23: ALE cancer text

Sandra has had 10 sexual partners. Based on the model, she is more at risk to have cervical cancer than her friend who has had 15.

True

False

Don't know

Figure A.24: ALE cancer question

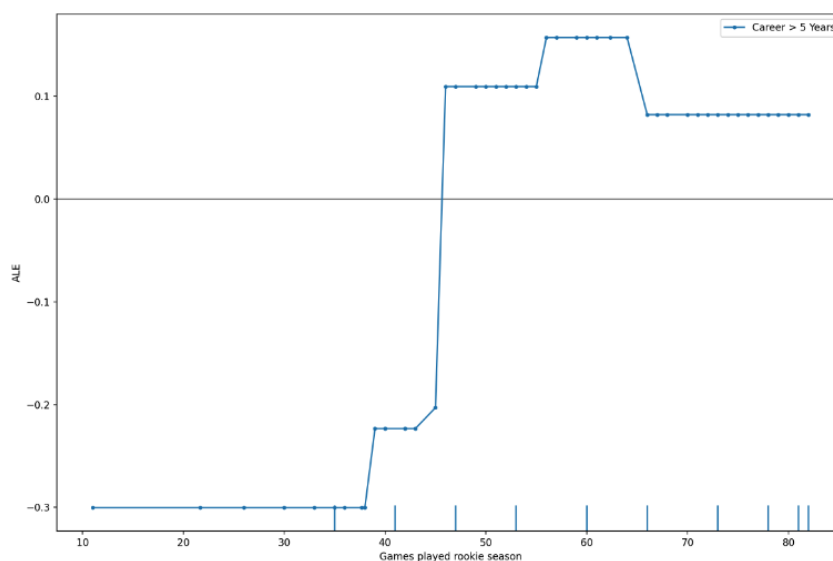


Figure A.25: ALE NBA visual

Overall, the average marginal effect of the model predicting "Career > 5 Years" based on "Games played rookie season" is between -0.301 and 0.162.

The feature "Games played rookie season" directly affects the model's outcome. The accumulated local effects of the feature "Games played rookie season" on the model predicting "Career > 5 Years" starts at -0.301. There is a lot of variability in the output based on the feature variable. The model is most likely to predict "Career > 5 Years" based on the feature "Games played rookie season" when the feature is at 59 with an accumulated local effect of 0.162. The model is least likely to predict "Career > 5 Years" when the feature has a value of 36 with an accumulated local effect of -0.301.

Figure A.26: ALE NBA text

Based on the model, if you are playing 36 games in your rookie season, then you are likely to have a long career.

True

False

Don't know

Figure A.27: ALE NBA question

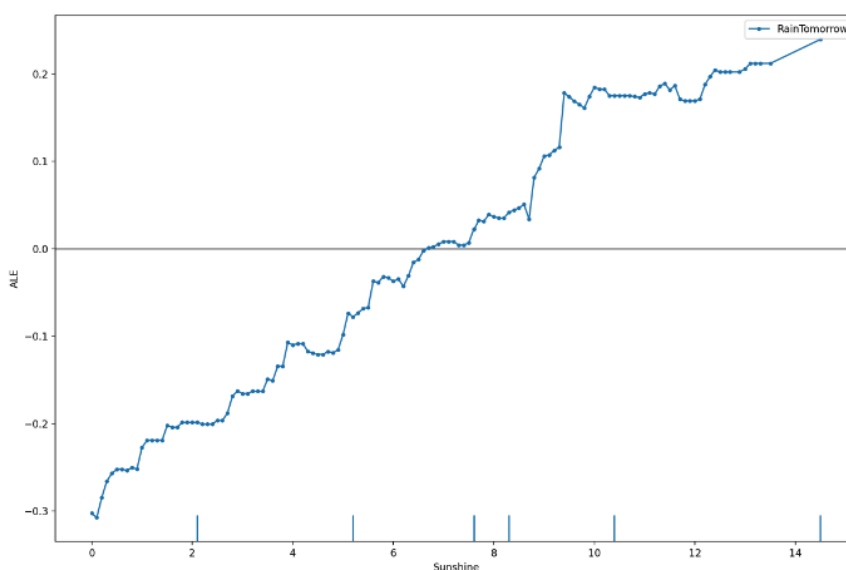


Figure A.28: ALE rain visual

Overall, the average marginal effect of the model predicting "RainTomorrow" based on "Sunshine" is between -0.309 and 0.239. The feature "Sunshine" directly affects the model's outcome. The accumulated local effects of the feature "Sunshine" on the model predicting "RainTomorrow" starts at -0.309.

There is a lot of variability in the output based on the feature variable. The model is most likely to predict "RainTomorrow" based on the feature "Sunshine" when the feature is at 15 with an accumulated local effect of 0.239. The model is least likely to predict "RainTomorrow" when the feature has a value of 0 with an accumulated local effect of -0.309.

Figure A.29: ALE rain text

George will go to the beach tomorrow, but only if it's not raining.

Today Sunshine has a low value of 0. Therefore, based on the model, it is likely that George will go to the beach tomorrow.

- True
- False
- Don't know

Figure A.30: ALE rain question