Explaining the auction gym doubly robust bidding estimator using LIME and Tree Surrogate Models with SHAP

Radhikesh Jain

Master of Science
School of Informatics
University of Edinburgh
2023

Abstract

Models are becoming more and more complex and less interpretable. They essentially are black boxes to us. Explainable AI aims to make the models more interpretable, thus increasing trust and making the models more transparent. Auction Gym [6] is an online auction simulation environment that maximises the bidding agent's utility. The doubly robust bidding estimator estimates the bid value to place based on the context. Various auction settings can be simulated in the auction gym. Auctions vary in competitiveness, ranging from low participant numbers to high participant numbers in a given round. There can be a varying number of features known to the bidding agent. The more the number of features, the more complex the decision-making becomes for the auction gym for placing an optimal bid value for the agent.

The aim is to make this complex bidding estimator more transparent using the explainable techniques in the literature. Techniques like Tree Surrogate Models (Decision Trees and Random Forest) with SHAP and LIME on the doubly robust estimator explain the complex auction gym bidding agent estimator for various auction settings.

It was observed that tree-based models, which are less complex and easy to interpret by humans, were able to mimic the complex doubly robust estimator quite well for settings of less competition and less number of contextual features. This could be observed as they got a high R2 score for these settings. As the number of features increased, the bidding pattern of the complex model became harder to mimic, which could be reflected in their lower R2 scores.

Tree SHAP and LIME outcomes revealed that all the explainable methods showed similarities in both 4-feature and 12-feature scenarios. They effectively identified feature importance based on the magnitude of the linear ground truth weight vector, with only slight variances in ranking features of close importance across methods. Interestingly, if the weight sign of a feature changed, the explanation's direction remained unchanged. This suggests that the auction gym model is more intricate than a linear model.

It was also observed that the computation of feature importances for surrogate tree-based models was much faster than the computation of the feature importances by LIME on the auction gym model. Also, Fast Tree SHAP [19] applied on Random Forest significantly improved the computation time of SHAP values compared to the regular Tree SHAP.

Research Ethics Approval

This project was planned in accordance with the Informatics Research Ethics policy. It did not involve any aspects that required approval from the Informatics Research Ethics committee.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Radhikesh Jain*)

Acknowledgements

I am thankful to our university supervisor Iain Murray for giving me the opportunity to work on this project. I am also grateful to our supervisors Ben Allision, Robert Hu, and Doudou Tang at Amazon for taking out their precious time and guiding me throughout my thesis right from my Informatics Project Proposal. Also, special thanks to Robert Hu and Doudou Tang for helping me with my doubts after IPP during our weekly Thursday meetings. I also have my sincere gratitude to Jingxuan Chen, Keith Wu, and Miltiadis Chrysopoulos, who worked on different aspects of the project. There was always a good discussion with each and everyone during our weekly meetings. Lastly, I would also like to thank all my friends and family who supported me throughout the challenging times of my dissertation.

Table of Contents

Chapter 1

Introduction

1.1 Motivation of the Project

Models are becoming more and more complex and harder to understand. They are essentially black boxes to humans. They are not transparent and are difficult to trust. The problem becomes more common in domains like medicine, where trust and transparency are of utmost importance, as it can become a matter of life and death for a patient. Explainable AI (XAI) aims to make the model more transparent and trustworthy by explaining the reasoning behind the model's output.

Most of the advertisements are held through an online auction mechanism. It is the major source of income for companies like Google and Facebook. Also, the companies who want to advertise must strategise techniques to win the right to advertise. It can be essential to win an auction as it can increase their revenue. The problem is that auction data is not readily available and is expensive to obtain. Olivier Jeunen et al. [6] introduced an Auction gym Reinforcement Learning environment designed to generate auction data through a simulation environment. They also introduced a novel Doubly Robust estimator for bidding to maximise revenue for individual bidders. In an auction, the bidder has the contextual features to work with. Based on this information, it decides which advertisement it wants to display from its inventory of advertisements. After that, the bidder has to decide the bid amount it wants to place based on the context information and the advertisement it decided to show. The Doubly Robust bidding estimator estimates the optimal bid. Thus, given contextual information, this tool helps create a strategy and estimate the optimal bid amount to win an auction. However, as a complex estimator, its decision-making process is essentially a black box to us. Therefore, we aim to make this auction gym model more transparent by using the XAI

techniques in the literature.

1.2 Aims and Objectives

The primary aim of this thesis is to explain the Doubly Robust bidding estimator in the Auction Gym environment. To achieve this, we employ techniques like surrogate models, decision trees, and random forests to create interpretable approximations of the complex auction gym model's behaviour. Moreover, we use SHAP (SHapley Additive exPlanations) to analyse feature importances obtained from the surrogate models. Additionally, we apply LIME (Local Interpretable Model-agnostic Explanations) as another XAI technique to gain insights into the decision-making process of the Doubly Robust estimator itself. Furthermore, we will analyse the impact of perturbing features on the resulting bid and ensure the coherence of explanation methods with established ground truth.
The specific objectives of this research are as follows:

1. Evaluate the performance of surrogate models across various auction settings, considering factors such as the number of contextual features, number of agents, and auction competitiveness.

2. Assess the similarity and alignment of feature importance of the explainable techniques with the established ground truth. Additionally, evaluate the consistency of these explanations in capturing the impact of feature perturbations on the Doubly Robust bidding estimator's decision-making process.

3. Assess the time taken by different XAI techniques for explaining the Doubly Robust bidding agent.

1.3 Research Questions

These are the following research questions being addressed in the research.

1. How effectively do surrogate models, including decision trees and random forests, approximate the complex behaviour of the Doubly Robust bidding estimator in different auction settings?

2. How similar are the feature importance insights from SHAP values of surrogate models with LIME explanations? Furthermore, do they align with the established ground truth?

3. How does the time taken for the various XAI techniques compare in explaining the decision-making process of the Doubly Robust bidding estimator?

1.4 Structure

The paper is structured into multiple sections. Chapter 2 discusses the background and literature review of explainable AI techniques and auction gym. In Chapter 3, the methods of surrogate models and LIME for explanation are discussed in detail. In Chapter 4, we present the results of the experiments for various auction gym settings for the above techniques. Finally, in Chapter 5, we conclude our findings and provide further work for the research.

Chapter 2

Background

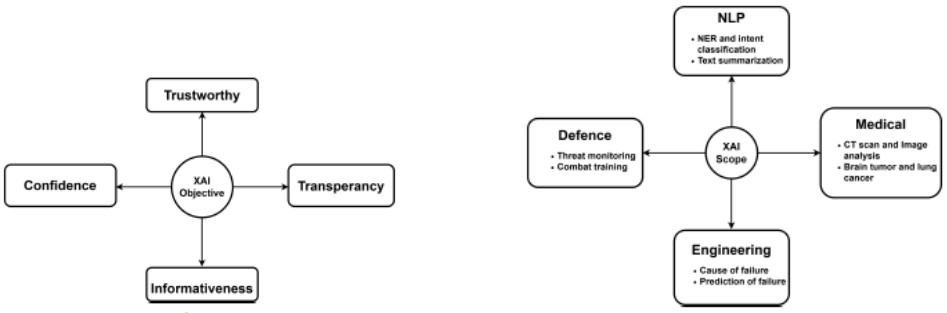
In this chapter, we will first discuss the background of explainable AI (XAI) and the techniques employed in the literature. Further, a background discussion about the auctions and the auction gym environment will be done.

2.1 Introduction to Explainable AI

2.1.1 XAI Scope and Objectives

Models are becoming increasingly complex, and they are black boxes to us. To be able to trust their decisions, we have to be able to interpret why they are arriving at their decisions. Explainable AI is the field that focuses on interpreting complex models like neural networks and Reinforcement Learning algorithms. Explainable AI (XAI) aims to bridge the gap between the opacity of AI models and the need for human-understandable explanations in various real-world applications, particularly in fields like healthcare, banking, and autonomous systems. By revealing how AI models arrive at their conclusions, XAI provides transparency and interpretability to AI systems. By doing so it aims to give more information behind the model's decisions. The XAIs objectives mentioned by Gohel et al. [5] can be seen in Figure 2.1a.

XAI is very useful for interpretability and has been useful in many domains. There is a wide scope of Explainable AI, as seen in Figure 2.1b. It is very crucial in the healthcare domain where the models can significantly impact patient trust and safety. Doctors will get a more informed analysis of why the machine learning model reached a particular conclusion, thus being more transparent and which can be easily trusted. Layer-wise relevance propagation was employed by Böhle et al. [3] to understand deep



(a) Objectives of XAI (b) Scope of XAI

Figure 2.1: XAI a tool to uncover black box models [5]

neural network judgments in MRI-based Alzheimer's disease classification, revealing insights into the important features utilized by the model for diagnosis.

It is also useful in the field of finance, where understanding the rationale behind model predictions is vital for risk assessment, fraud detection, investment decisions, and understanding the stock market. Jean Jacques Ohana et al. [11] aims at enhancing the interpretability of AI models used in financial markets, allowing market participants and regulators to understand the decision-making processes of these models better and gain insights into market behaviour.

It is also very important in the field of autonomous driving to gain trust in the ability of driverless vehicles. Explainable AI approaches to aid in making autonomous system decisions more transparent and understandable, fostering trust in the security of their operation. Atakishiyev et al. [2] do a comprehensive overview of the explainable techniques in the field of autonomous driving.

2.1.2 XAI Methods

There are many ways to interpret a black box model. Some of the techniques are model dependent, like TreeSHAP for tree-based models used in [8], while others are model agnostic like LIME [12], Kernal SHAP [7], and RKHS SHAP [4]. Model-dependent techniques are specific to a particular model, while model-independent techniques apply to any black-box model. Explainable AI (XAI) offers both local and global interpretability. On a local level, it clarifies individual predictions by determining the contribution of each feature to specific model output. On a global scale, it delivers insights into feature importance across the entire dataset, highlighting the factors most significantly influencing the model's overall predictions. Figure 2.2 provides a pseudo

ontology of XAI methods taxonomy mentioned by Amina Adadi et al. [1]

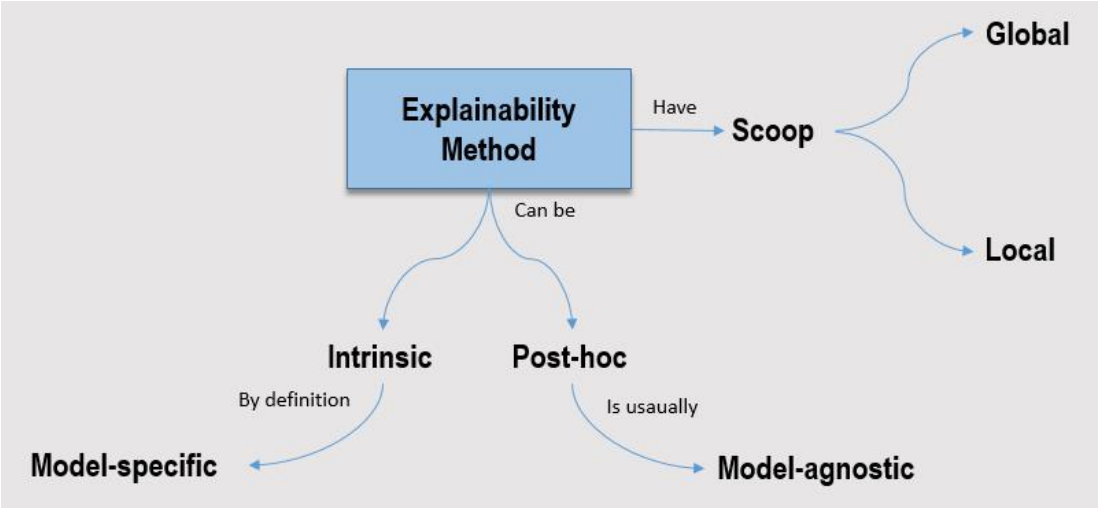


Figure 2.2: A pseudo ontology of XAI methods taxonomy [1]

SHAP (SHapley Additive exPlanations) is a prominent method for explaining the predictions of complex machine learning models. Developed by Lundberg et al. [7], SHAP is based on cooperative game theory and utilizes the Shapley value [13] concept for fairly distributing each player's contribution in a cooperative game.

In the context of Explainable AI (XAI), SHAP is employed to measure the contribution of each feature to a specific model prediction. It assigns SHAP values, which are importance values, to individual features, indicating their impact on the model's output. These values represent a fair and consistent way to attribute the model's prediction to each feature, considering all possible feature combinations.

SHAP values offer both local and global interpretability. Locally, SHAP explains the prediction of a particular instance by demonstrating how each feature contributes to that specific prediction. Globally, SHAP values provide an understanding of feature importance across the entire dataset, highlighting the features that have the most significant impact on the model's overall predictions.

LIME, introduced by Ribeiro et al. [12], is a widely-used method for explaining the predictions of complex machine learning models. LIME is designed to work with any black-box model and generate local explanations for individual predictions. It approximates the complex model with a simpler, interpretable model in the local neighbourhood of a specific instance. By perturbing the input data, observing the changes in the model's predictions, and fitting a surrogate model that mimics the behaviour of the original model locally, LIME provides insights into how the features

contribute to the prediction for that particular instance.

Another way the complex model is explained is by using a less complex model, also known as a surrogate model, which mimics the behaviour of the original complex model. One such surrogate model used is trees. Sieusahai et al. [15] used trees as a surrogate model to explain the reinforcement learning agents in the Atari domain.

Surrogate models, such as decision trees, are trained to approximate the predictions of black-box models. By doing so it allows us to draw conclusions about the black-box models by interpreting the surrogate models. These models can be easily interpreted by visualizing the decision tree structure. We can gain insights into the decision-making process and understand the factors contributing to the predictions. Decision trees are widely used for feature importance analysis in linear and non-linear models, making them suitable for global explanations in XAI. They provide insights into the behaviour of the AI models and can be used for both global and local explanations.

Notably, recent research has actively explored the realm of explainable techniques for neural networks. DeepLift [14] employs a technique that propagates differences through the network to discern significant features. Similarly, integrated gradient techniques [17] have been applied to unveil neural network workings. However, compared to neural networks, the exploration of explainability within the context of RL models has been relatively limited. Sieusahai et al. [15] use XAI techniques on Deep Reinforcement Learning Agents in Atari games by employing surrogate models.

Speith et al. [16] mentioned the result-based approach proposed by McDermid et al. [9]. The result-based approach can be seen in Figure 2.3. It presents a systematic

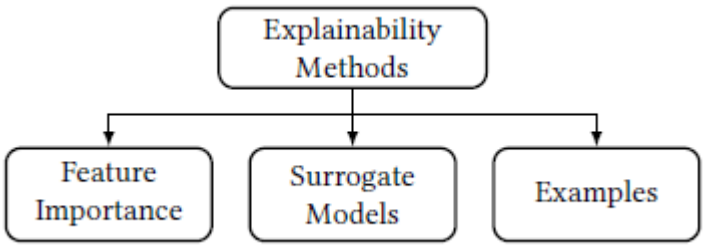


Figure 2.3: Result based approach proposed by [9] for XAI

classification of explainability methods centred on the outcomes they produce. This classification helps users choose the right method that aligns with their application's specific requirements. The taxonomy consists of three main categories:

1. Feature Importance: This category emphasizes methods that demonstrate the

impact of input features on model outputs, offering insights into the importance of features.

2. Surrogate Models: These models approximate complex models with simpler, interpretable versions. They help users understand complex models and can be created using various techniques.

3. Examples: Explanations are provided through representative examples, showcasing instances that lead to high or low-certainty predictions. This category delivers intuitive insights into model behaviour.

The Result-Based Approach supports users in selecting appropriate explainability methods based on their expertise and the desired level of explanation.

2.1.3 XAI Evaluation

There is a concern about evaluating the results of the explainable techniques. For a surrogate model like decision trees, one can compute the accuracy score (for classification task) or compute the R2 score (for regression task) and see the performance of the model as done by Andreas Messala et al. [10]. In the case of SHAP and LIME, one can give more importance to some of the features and check the variations of the results. Ideally, a slight change in the less important feature should not affect the output significantly, while changing the more important feature should affect the output by a drastic amount. Another way of interpreting the outputs is that they are analysed by human experts who confirm whether the results of explainability techniques make sense.

Moreover, there is also concern about the speed of explainability techniques. It can be very time-consuming to generate explanations for complex models compared to simpler ones. There have been approaches to reduce the time taken for the explanations. One such instance is using Fast Tree SHAP proposed by Yang et al. [19] to compute Shapley values for tree-based models. Although the Fast Tree SHAP v1 proposed in the paper and the original Tree SHAP have the same time complexity of $O(MLTD^2)$, the Fast Tree SHAP v1 reduced the average running time by 25%. Figure 2.4 shows the reduced time Fast Tree SHAP takes to generate SHAP values.

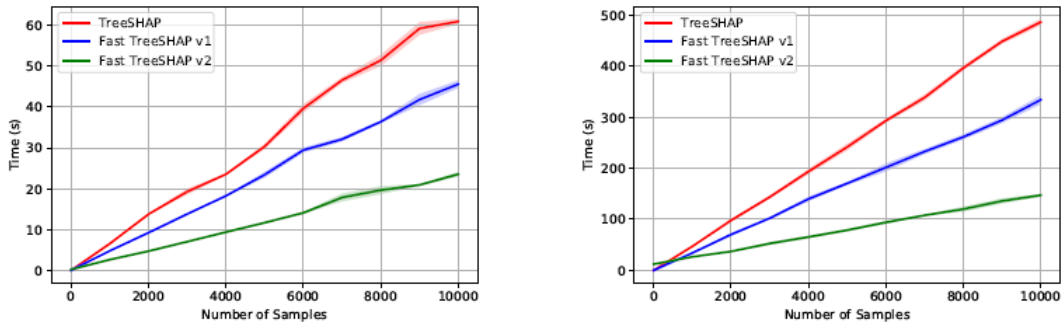


Figure 2.4: [19] Showing effectiveness of Fast TreeSHAP

2.2 Auctions and Auction Gym

Auctions have been widely studied across various disciplines, such as economics, game theory, and computer science. The study of auctions can be traced back to the seminal work of Nobel laureate William Vickrey, who introduced the concept of a second-price sealed-bid auction, also known as a Vickrey auction [18]. Since then, numerous auction formats have been proposed and analysed, each with its unique strengths and weaknesses.

Traditional auction theory focuses on theoretical models and analytical solutions for optimal auction design. However, applying these theoretical models to real-world scenarios can be challenging due to complex interactions between bidders, imperfect information, and strategic behaviour. This gap between theoretical models and real-world applications led to the development of simulation environments like Auction Gym [6].

Auctions are crucial for resource allocation, pricing, and revenue generation in various industries. However, applying AI techniques in auction environments presents unique challenges due to the complexity of bidder strategies, allocation mechanisms, and auction outcomes. Interpretable AI in auction scenarios can help understand and explain the decision-making process behind bidding strategies and auction outcomes. By making AI-driven auction systems more transparent, stakeholders, including auction participants, regulators, and market designers, can gain confidence in the mechanisms, identify potential biases, and make informed decisions for optimizing the auction process. This can improve the trust and accountability of AI systems and enable better collaboration between humans and AI.

Auction gym [6] is an open-source online auction simulation environment created

by Amazon. Offline auction data is not readily available due to the confidentiality of the auction, and online data is costly and not feasible to obtain. Thus, it benefits the research community, in general, to generate auction data via the reinforcement learning simulation environment.

Auction Gym is a simulation platform that enables researchers to conduct experiments and research in auction scenarios. It provides a controlled environment where researchers can simulate and test various auction types, bidder strategies, and allocation mechanisms. The platform allows researchers to define custom configurations, such as the number of participants, items, and allocation rules, to create diverse auction scenarios resembling real-world situations. Auction Gym enables systematic evaluation of different bidding algorithms and allocators under controlled conditions, enabling researchers to gather valuable insights into bidder behaviour and the impact of various auction parameters. The platform has gained significance in the research community due to its ability to bridge the gap between theoretical auction models and real-world applications, fostering the development of more effective and transparent auction mechanisms.

The Auction Gym framework is designed to tackle two main problems: ad allocation and bidding. In the ad allocation problem, the agent (bidder) selects the most suitable advertisement from their inventory based on a given context of features. Once the advertisement is chosen, the agent determines the bid amount to place, considering the context features and the selected advertisement. There are various techniques used by the agent to learn how to place the optimal bid. The paper has proposed the novel Doubly Robust Estimator as a bidding technique which is shown to be effective.

XAI techniques have not been applied to the auction setting. In explaining the Auction Gym environment, XAI techniques can be employed to explain the behaviour of auction mechanisms, bidder strategies, and allocation decisions. By utilizing the XAI techniques, researchers can better understand the factors influencing auction outcomes and the impact of different features on the bidding process.

Applying XAI techniques to the Auction Gym context can help users understand the rationale behind individual bidding decisions made by agents. This can lead to identifying potential issues, such as model bias, robustness, and causality, and provide valuable insights into the decision-making process in auction scenarios.

Chapter 3

Methodology

In this chapter, we provide the details of the methods employed in the project to explain the auction gym model. It is divided into four different sections. In the first section, discussion is done about the design choices. The other sections discuss how the data is generated, the techniques used for explanation, and the evaluation techniques used in the research.

3.1 Design Choices

The auction gym has various parameters that need to be set to run a particular auction environment. We will explain the parameters below and the values we chose to run the experiments.
The following are some of the parameters that need to be set for an auction environment:

1. **num_iter:** The agents in the auction gym environment update their policy after every iteration. Initially, the policy will not be optimal, and agents will try to learn via multiple auction rounds in each iteration. For the experiments, we have chosen the number of iterations as eight as the agents will be able to reach the optimal policy. Increasing the number above increases the time of experiments and has not been tried due to time constraints.

2. **rounds_per_iter:** This is the number of auction rounds held per iteration. This number is varied in our experiments based on the number of participants in an auction round and the number of context features considered in an auction round. The higher the number of features and the higher the number of participants in an auction round, we increase the number of auction rounds held per iteration.

3. **Embedding size:** Embedding size signifies the number of context features that are there in the environment, and observed embedding size is the number of context features known to a bidder before deciding the advertisement to choose from its inventory and the bid amount to place for the given auction setting. For our experiments, we have considered the observed embedding size as 4, 8, 12, 16, and 20 while the embedding size is one more than the observed embedding size, i.e. 5, 9, 13, 17, and 21. The default features in the auction gym are normally distributed. For the experiments where the number of observed features is 4, 8, and 12, all the features considered are normally distributed. However, for observed features of 16 and 20, a mixed variety of features are considered. When the number of observed features is 16, there is a total of 12 normally distributed features with two uniformly distributed and two binary features. Of 20 observed features, 12 are normally distributed features with four uniformly distributed and four binary features.

4. **Allocation:** In the auction gym allocation can be of two types: First Price and Second Price. Second Price auctions are easy to win by truthfully bidding [18] and are not beneficial for the auctioneer. Thus, most of the auctions held are First Price. Due to this, only First Price auctions are considered in our experiments.

The following are some of the parameters that need to be set for an agent in the auction environment:

1. **Allocator:** There are two types of allocators in the auction gym environment: OracleAllocator and PyTorchLogisticRegressionAllocator. The OracleAllocator is an ideal allocator that has perfect knowledge of the bidder's valuations for the items. It knows the true underlying valuations of each bidder for all the items. It is not useful for real-world scenarios. On the other hand, PyTorchLogisticRegressionAllocator leverages observable context and historical data to estimate bidder valuations, making it more applicable to real-world auction scenarios. Hence, we chose to work with PyTorchLogisticRegressionAllocator for conducting our experiments.

2. **Bidder:** Bidders in the AuctionGym environment can adopt four strategies: TruthfulBidder, ValueLearningBidder, PolicyLearningBidder, and DoublyRobustBidder. For our experiments, we chose the DoublyRobustBidder policy to work with due to its potential robustness, optimality, and ability to combine value

learning and policy learning methods. It has been proven an effective method in the auction gym paper [6].

3. **num_copies and num_items:** Specifying the number of agent copies creates multiple agents with the same configuration within the auction environment. The 'number of items' field indicates the quantity of ad catalogues in an agent's inventory. We selected an ad catalogue containing 12 advertisements for the bidding agents for our experiments.

An example of the JSON file with the above-mentioned parameters can be seen in Figure 3.1

```
{
  "random_seed": 0,
  "num_runs": 3,
  "num_iter" : 8,
  "rounds_per_iter": 10000,
  "num_participants_per_round": 3,
  "embedding_size": 5,
  "embedding_var": 1.0,
  "obs_embedding_size": 4,
  "allocation": "FirstPrice",
  "agents": [ {
    "name": "DR",
    "num_copies": 10,
    "num_items": 12,
      "allocator": {
        "type": "PyTorchLogisticRegressionAllocator",
        "kwargs": {"embedding_size": 4, "num_items": 12}
      },
      "bidder": {
          "type": "DoublyRobustBidder",
          "kwargs": {
            "gamma_sigma":  0.02,
            "init_gamma": 1.0
          }
    }
  }],
  "output_dir": "results/FP_DR_TS/"
}
```


Figure 3.1: JSON file having the configurable parameters for the auction gym

3.2 Data Collection from Auction Gym Model

The Doubly Robust bidding policy generates data from the auction gym reinforcement simulation environment. The JSON configuration file for a particular setting is created, and the model is first trained. After the model is trained for the set iterations in the configuration file, the data is generated for each agent in the environment and stored in the local machine. The features are the context available to the bidding agent. The

number of features available to the bidding agent equals the observed embedding size. The label is the bid value the agent bids given a particular context, which the doubly robust estimator learns. Moreover, the trained model for each agent in the auction environment is saved in the local machine, which can be used later for further analysis.

3.3 Techniques

3.3.1 Tree Surrogate Models (Decision Trees and Random Forest)

The first technique that we employ is tree-based models as surrogate models. For this purpose, we use decision trees and random forests to mimic the complex Doubly Robust bidding estimator in the Auction gym environment. Surrogate models try to mimic the entire complex model and thus give a global explanation of the bidding mechanism. 50000 data points are sampled from the saved dataset obtained after training the auction gym model for each experimental setting. The sample is done by taking random_state as 18. These 50,000 data points are then split into training and test datasets by keeping 20% of the data points in the test data. Grid Search is used to find the best hyperparameters for the decision trees by varying some hyperparameters. Below are the parameters and their values considered while doing a Grid Search CV to find the best decision tree.

- **max_depth:** None, 5, 10, 15, 20
- **min_samples_split:** 2, 5, 10
- **min_samples_leaf:** 1, 2, 3, 4, 5

Also, cross-validation of 5 is taken while doing GridSearchCV to find the best decision tree. If the max depth of the tree is not restricted, it overfits by giving a very high train accuracy score while performing very poorly on the test and unseen data.

Random Forest is an ensemble of decision trees and thus can have better accuracy than decision trees for the same experimental setting. However, the random forests can be more complex, reducing the model's interpretability. Visualisation of the decisions by a decision tree will be more easily interpretable by humans.

3.3.2 Tree SHAP and Fast Tree SHAP on Tree Surrogate Models

For the tree surrogate models of Decision Trees and Random Forest Tree SHAP is used to compute the feature importance of the contextual features. 200 data points are

selected from the test dataset, and the mean absolute SHAP value is computed for each and every feature to provide a global interpretation based on the 200 data points. Fast Tree SHAP is also used to compute the SHAP values for the Random Forest Model, as they are more complex and time-consuming to explain.

3.3.3 LIME

The other technique which has been used for explaining the auction gym model is LIME. For implementation, the LIME library in Python is used. LIME is an agnostic model that can be applied to any model (decision trees, neural networks, RL agents, etc.). It provides an explanation that is locally accurate and situated in the neighbourhood of the observation or example that is being described. The process generates 5000 samples for the feature vector as a default setting, following a normal distribution. Subsequently, it acquires the target variable for these 5000 samples using the prediction model that is the focus of the explanation. After obtaining the surrogate dataset, it weighs each row according to how close they are to the original sample/observation. It then uses feature selection techniques to find the most important features. In our experiments, LIME is used on the doubly robust bidding estimator of the auction gym model itself by taking 200 sample data points like those employed for decision trees and random forests. The mean absolute LIME feature importances for these data points are then computed for different experimental settings.

3.4 Evaluation

3.4.1 Accuracy Metric

As the task of predicting the bid the agent places given a context is a regression task, the R2 score is a good metric to evaluate the performance of the surrogate models. Therefore, decision trees and random forest regressors are evaluated by using the R2 score. This is performed on 50,000 data points for various experimental settings. The performance of decision trees and random is checked and compared by changing the number of features and the number of participating agents in an auction round. R2 score is also checked by increasing the number of agents in the environment and keeping the number of participants in an auction round constant for a varying number of context features. The performance is also checked for multiple agents in a multi-agent auction

environment to evaluate the consistency of the performance of the tree-based models across agents.

3.4.2 Feature Importance Evaluation

The feature importance is seen using Shapley values for surrogate models decision trees and random forests. Python's shap library is used to compute the Shapley values. Similarly, LIME is used to see the feature importances for 200 data points directly on the doubly robust bidding agent of the auction gym environment. To achieve this, Python's lime library finds the feature importances. For all the methods, the mean absolute of the feature importances is computed for the 200 data points for particular experimental settings and compared. To have a deeper understanding of the feature's importance, a graphical visualisation and beeswarm plot is also used to compare the results by the three methods. A beeswarm plot effectively displays an information-dense summary of how the top features in a dataset impact the model's output.

Further, the explanation results are checked to see if they are consistent with the established ground truth. This is done by multiplying the input features with a known ground truth weight and observing if the explanation methods produced what was expected. Furthermore, each feature is perturbed individually, and the resulting mean absolute change in the output is observed as an outcome of altering each feature. Ideally, a more important feature in the explanation should induce a greater change in the output compared to a less significant feature. If they align then it would further reinforce the reliability of the explainability methods. Finally, some local explanations are observed with the help of LIME to see if the explanations are consistent with the explanations provided by the 200 data points.

3.4.3 Time Evaluation

Explanations can take a long time if the model is complex. Thus, the explanation is evaluated by computing the time taken by the explanation methods used: Tree SHAP on Decision Trees, Tree SHAP on Random Forest, Fast Tree SHAP on Random Forest and LIME on the Doubly Robust bidding agent. Time taken to compute Shapley values on the surrogate models of decision trees and random forest and time taken to compute LIME explanation on the auction gym model is observed. To compare the time taken between the three, 200 data points are considered, and features are varied as 4, 8, 12, 16, and 20. The effect of time taken to compute the explanation is observed with

the increase of features. Moreover, the effect of time on increasing the data points is observed. Python's time library calculates the time taken in seconds for the explanation methods.

Chapter 4

Experimental Results

In this chapter, we provide the details of the results obtained in this project. It is divided into three different sections. In the first section, a discussion is done about the R2 scores obtained by the surrogate models under different auction settings. In the second section, the feature importance of different techniques is discussed and compared with the expected ground truth. In the third section of experimental results, time taken by various explainability techniques is analysed and discussed.

4.1 R2 Score Decision Trees and Random Forest

In this section, we discuss the experimental results obtained from using decision trees and random forests as surrogate models to explain the behaviour of the Doubly Robust auction gym RL model. The experiments for fitting the surrogate models were conducted using a large dataset of 50,000 data points, considering various experimental auction settings to assess the performance and interpretability of tree-based models.

Firstly, the impact on the R2 score by varying the number of agents in the auction environment while keeping the number of participants per auction round constant as two was analysed. It was observed that the R2 test score for a particular number of contextual features remains very close even when the number of agents increases in the environment. This is seen for both decision trees and random forests and can be seen in Table 4.1.

It can also be observed from the table that the surrogate models become less effective in mimicking the auction gym model when the number of features increases. However, they remain consistent when the number of agents in the environment increases for a fixed contextual feature.

num_features	num_agents				
	4	**6**	**8**	**10**	**12**
4	0.937	0.873	0.917	0.934	0.940
8	0.593	0.593	0.629	0.657	0.654
12	0.382	0.317	0.355	0.349	0.383
16	0.320	0.310	0.324	0.323	0.395
20	0.278	0.227	0.270	0.307	0.324

(a) Test accuracy (R2 score) for decision trees

num_features	num_agents				
	4	**6**	**8**	**10**	**12**
4	0.959	0.912	0.961	0.971	0.979
8	0.810	0.807	0.823	0.838	0.835
12	0.634	0.584	0.626	0.614	0.669
16	0.546	0.553	0.543	0.537	0.632
20	0.523	0.501	0.483	0.533	0.563

(b) Test accuracy (R2 score) for random forest

Table 4.1: Test accuracy for number of participants per round as 2 and varying number of agents in the environment

Next, we conducted experiments by varying the number of participants per round (competition) while keeping the number of agents constant at 12. Additionally, the number of features in the auction gym environment varied between 4, 8, 12, 16, and 20. The results indicated that tree-based models performed well for experiments when the number of features was less. As the number of features increased from 4 to 12, the performance of tree-based models reduced, but there was not much difference when the features increased to 16 and 20. This is because the features added were uniformly distributed and binary, which did not affect the R2 score drastically. Generally, the tree-based models found it more difficult to mimic the doubly robust bidding estimator with the increased number of features. However, Random Forests had a much better R2 score in settings where the number of features was more compared to decision trees. This trend can be seen in Figure 4.1 shows the decrease of test R2 score with the increase in the number of features for both decision trees and random forests where the number of participants in an auction round is 4.

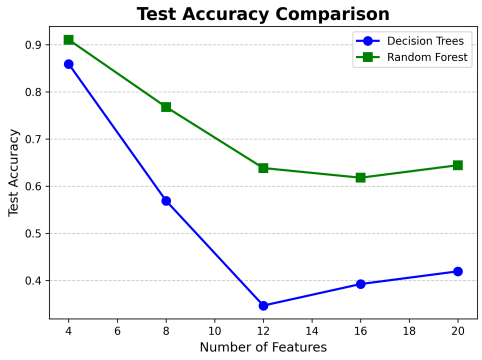


Figure 4.1: Performance of Surrogate Models for num participants as four and number of agents as 12

Additionally, the trend of the increase in the number of participants in an auction round was observed. It was seen that there was a slight decrease in the R2 score as the competitiveness of the auction increased. This trend was true for both decision trees and random forests. The tree-based models had better R2 scores when the number of participants per auction round was 2 in comparison to when the number of participants per auction round was 6 for features 4, 8, 12, and 16. However, for the context feature of 20, the R2 score was higher when the number of participants was 6 in comparison to 2 participants per auction round. The apparent slight increase in R2 scores may be because of the already low R2 score. Hence, further research with a significant increase in the number of participants would help understand the effect on R2 scores with an increase in the number of participants per round for a high feature setting. The findings of the R2 score for these various settings can be seen in Table 4.2 for test data. Table 4.3 shows the training R2 scores of the tree-based models.

num_participants	num_features				
	4	**8**	**12**	**16**	**20**
2	0.940	0.654	0.383	0.395	0.324
3	0.859	0.662	0.337	0.309	0.380
4	0.859	0.569	0.347	0.393	0.420
5	0.723	0.590	0.368	0.404	0.397
6	0.780	0.554	0.363	0.345	0.436

(a) Test accuracy (R2 score) for decision trees

num_participants	num_features				
	4	**8**	**12**	**16**	**20**
2	0.979	0.835	0.669	0.632	0.563
3	0.912	0.840	0.599	0.535	0.613
4	0.911	0.768	0.639	0.618	0.645
5	0.796	0.783	0.630	0.605	0.600
6	0.853	0.758	0.608	0.553	0.673

(b) Test accuracy (R2 score) for random forest

Table 4.2: Test accuracy for varying features and number of participants in an auction round

num_participants	num_features				
	4	**8**	**12**	**16**	**20**
2	0.995	0.913	0.519	0.524	0.456
3	0.951	0.904	0.468	0.448	0.501
4	0.954	0.834	0.489	0.537	0.609
5	0.865	0.840	0.504	0.541	0.535
6	0.913	0.813	0.515	0.516	0.613

(a) Train accuracy (R2 score) for decision trees

num_participants	num_features				
	4	**8**	**12**	**16**	**20**
2	0.997	0.976	0.952	0.949	0.939
3	0.988	0.977	0.944	0.935	0.945
4	0.988	0.967	0.949	0.945	0.950
5	0.971	0.969	0.947	0.944	0.944
6	0.980	0.966	0.945	0.938	0.954

(b) Train accuracy (R2 score) for random forest

Table 4.3: Train accuracy for varying features and number of participants in an auction round

Furthermore, the R2 score was analysed across multiple bidding agents to check the consistency of the surrogate models in the multi-agent reinforcement auction gym

setting. This was done for the setting of 12 agents in the environment. The number of participants in the auction gym environment was kept as 4, and it was analysed for features as 4, 8, 12, 16, and 20. It was found that the performance of both decision trees and random forest remained consistent across all agents for a fixed number of features, which can be seen in Table 4.4.

num_features	**agent_id**											
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
4	0.911	0.964	0.909	0.932	0.973	0.948	0.919	0.976	0.861	0.879	0.877	0.889
8	0.768	0.804	0.757	0.700	0.819	0.789	0.800	0.821	0.832	0.839	0.783	0.818
12	0.639	0.554	0.639	0.666	0.581	0.624	0.617	0.677	0.639	0.549	0.601	0.619
16	0.618	0.512	0.602	0.658	0.665	0.564	0.602	0.584	0.646	0.472	0.598	0.615
20	0.645	0.579	0.488	0.657	0.467	0.566	0.599	0.598	0.452	0.614	0.547	0.521

(a) Test accuracy for random forest for different agents and different features

num_features	**agent_id**											
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
4	0.859	0.931	0.872	0.894	0.946	0.912	0.867	0.935	0.798	0.819	0.815	0.832
8	0.569	0.609	0.539	0.502	0.621	0.600	0.576	0.637	0.638	0.707	0.601	0.635
12	0.347	0.327	0.409	0.402	0.282	0.380	0.373	0.420	0.406	0.291	0.350	0.366
16	0.393	0.276	0.331	0.380	0.435	0.329	0.345	0.323	0.450	0.254	0.354	0.374
20	0.420	0.375	0.265	0.473	0.233	0.336	0.315	0.386	0.221	0.381	0.330	0.279

(b) Test accuracy for decision trees for different agents and different features

Table 4.4: Test accuracy for different agents and different features

Additionally, the correlation between features using 4 and 12 contextual features was examined to determine if it contributed to the decrease in the accuracy of the tree-based models. It was observed that the features showed no significant correlation, as evident from the heatmap of the correlation matrix in Appendix A. Figure A.1 illustrates the correlation between features when using four contextual features, while Figure A.2 displays the correlation between features when using 12 contextual features.

4.2 Feature Importances Surrogate Models and LIME

The feature importances were taken for all feature settings of 4, 8, 12, 16, and 20. The findings for contextual features 4 and 12 with the number of participants in an auction round as four are shown and compared in the report. This is because when the features were 4, the surrogate models had a high test R2 score of 0.86 for decision trees and 0.91

for random forest for the above setting. On the other hand, when the features are 12, the surrogate models start showing a lower R2 score, which is quite similar to when the features are 16 and 20. Decision trees have a test R2 score of 0.35, and random forest has a test R2 score of 0.64 when the features are 12 for the number of participants in an auction round as 4. Therefore, findings of contextual features of 4 and 12 are considered for visual purposes.

4.2.1 Feature Importance Analysis for Four Features

Firstly, four features are considered with all normally distributed features N(0,1) and ground truth weight vector as [1,1,1,1]. For this setting, it is seen that the mean absolute SHAP values on Decision Trees and Random Forest are very close to each other and are also quite close to the mean absolute LIME values for all four features. The mean absolute values for the three methods in this scenario can be seen in Table 4.5.

	Feature 1	**Feature 2**	**Feature 3**	**Feature 4**
Decision Trees	0.0628	0.0716	0.1034	0.0504
Random Forest	0.0623	0.0708	0.1046	0.0501
LIME	0.0814	0.0905	0.1332	0.0745

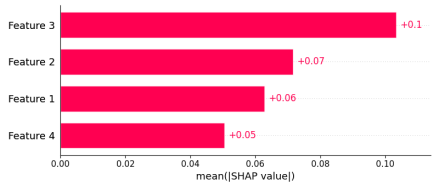
Table 4.5: Mean Absolute Values of Feature Importance for Four Features

This suggests that the three methods give similar explanations for the four feature setting. Also, the feature importances can be seen graphically in Figure 4.2.

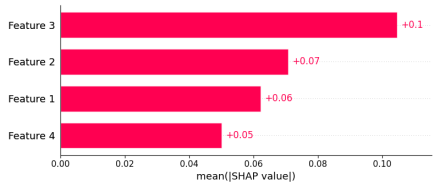
Next, a beeswarm plot is visualised to see the effect of features on the output bid. For the case of 4 features, one can see the beeswarm plot for the three methods in Figure 4.3.

It can be seen from the figure that all three methods convey the same thing for this particular experimental setting. The third context feature is the most important to make the decision.

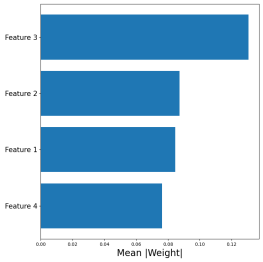
In a beeswarm plot, the higher value of SHAP/LIME favours a higher bid, while the lower LIME/SHAP value points to the direction of a lower bid. However, the influence of Feature 3 on bids is not straightforward, as its lower values occasionally align with higher bids and vice versa. In contrast, certain features exhibit more distinct bid impact patterns. Notably, all three methods consistently rank Feature 2 as the second most important, followed by Features 1 and 4. Importantly, the conclusions drawn from these methods are in harmony: elevated Feature 2 values, lower Feature 1 values, and higher



(a) Decision tree feature importances



(b) Random Forest Feature Importances



(c) LIME feature importances

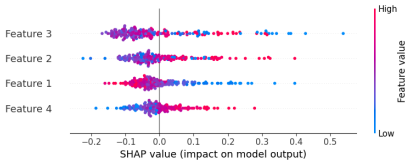
Figure 4.2: Feature importances for 4 features and 4 participants per auction round

Feature 4 values consistently correlate with higher bidding outcomes.

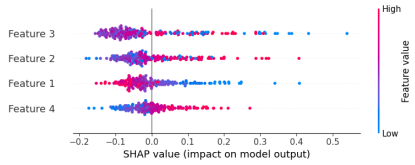
Next, the input features, drawn from a standard normal distribution N(0,1), are multiplied by a ground truth weight vector of [1, 1, 3, 0]. This weight assignment magnifies the significance of Feature 3 threefold while rendering Feature 4's weight as 0, effectively making it inconsequential. The expectation here is twofold: Feature 3 should emerge as the most influential, while Feature 4's impact should be negligible. Subsequently, the model is trained, and explanations are generated using all three methods.

The observations are encapsulated in the beeswarm plot depicted in Figure 4.4. Strikingly, all three explanation methods effectively capture this ground truth. The results corroborate the expectation: Feature 3 indeed emerges as the paramount contributor, significantly influencing bid values, while Feature 4 exhibits no discernible impact. Although the comparative importance of the remaining two features falls short of Feature 3, their relative significance to each other remains relatively consistent.

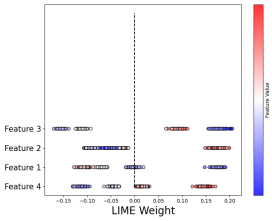
Further, to investigate the impact of sign changes, a ground truth weight vector, [-1, -2, 3, 0], was introduced. The hypothesis posited that reversing the sign of a feature's weight would trigger a corresponding reversal in its influence on the outcome if the model followed a linear relation. For instance, if a high value of a feature initially corresponded to high bid predictions with positive weights, it is expected that, with negative weights, lower values of the feature would now align with higher bid predictions. However, the observed results deviated from this anticipation. For instance,



(a) Decision tree beeswarm plot

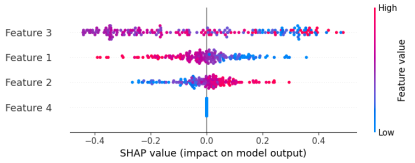


(b) Random Forest beeswarm plot

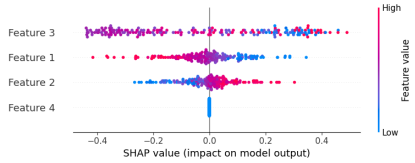


(c) LIME beeswarm plot

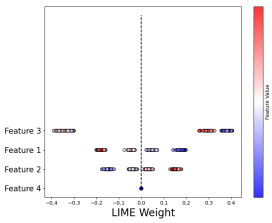
Figure 4.3: Beeswarm plot for 4 features with ground truth weight vector [1,1,1,1]



(a) Decision tree beeswarm plot



(b) Random Forest beeswarm plot

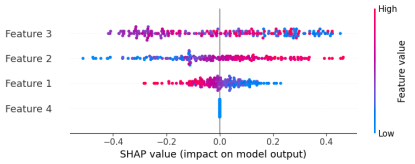


(c) LIME beeswarm plot

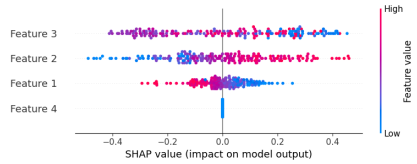
Figure 4.4: Beeswarm plot for 4 features with ground truth weight vector [1,1,3,0]

in Figure 4.4, Feature 1 carried a positive weight of 1, while Figure 4.5 bore a negative weight of 1. Astonishingly, the interpretation persisted unchanged: lower values of Feature 1 still correlated with higher bids. This phenomenon can be ascribed to the intricate dynamics of the auction environment. In contrast to the simplicity of a linear model, the intricate interplay of auction dynamics appears to mitigate the expected impact of sign changes.

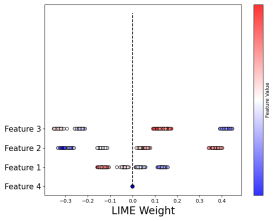
A perturbation of the features with the ground truth weight vector of [1,1,3,0] was also performed on the whole test data set to see the reliability of the explainable methods. Each feature was perturbed one by one by scaling it by two and observed how it changed the output bid by computing the mean absolute change in the output bid. It can be seen



(a) Decision tree beeswarm plot



(b) Random Forest beeswarm plot



(c) LIME beeswarm plot

Figure 4.5: Beeswarm plot for 4 features with ground truth weight vector [-1,-2,3,0]

from Figure 4.6 that the change in the output bid is the maximum when Feature 3 (most important) is perturbed, while Feature 4 does not cause much change in the output. This is in line with what the explainable methods convey.

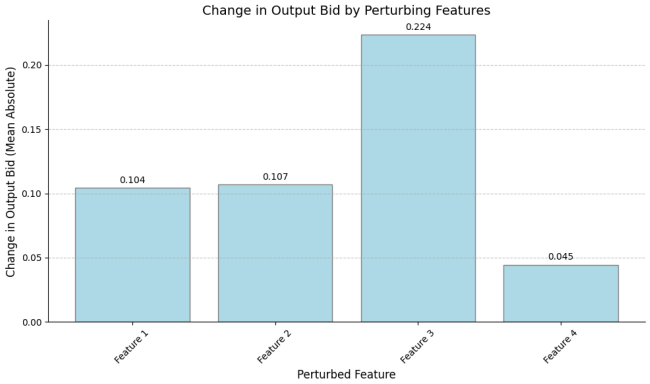


Figure 4.6: Effect in Output bid by perturbing for 4 features

4.2.2 Feature Importance Analysis for Twelve Features

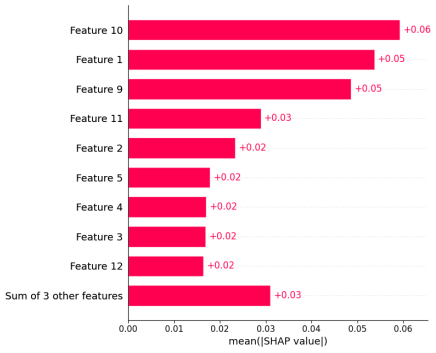
A further experiment was conducted with higher features of 12. Similarly to the four feature setting, first, a ground truth weight vector of [1,1,1,1,1,1,1,1,1,1,1,1] was considered, and the results of the three explainable methods were compared. It can be observed in Table 4.6 that for some features, the random forest has a mean absolute SHAP value closer to LIME than decision trees.

For example, for Feature 4, the absolute mean value for LIME is 0.0448, while for

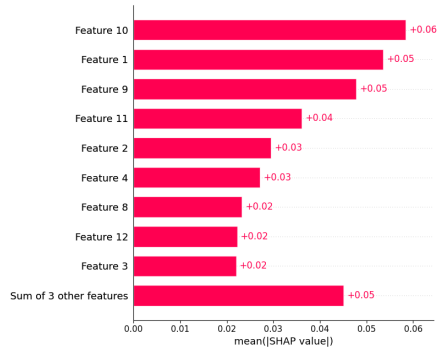
	Feature 1	Feature 2	Feature 3	Feature 4	Feature 5	Feature 6	Feature 7	Feature 8	Feature 9	Feature 10	Feature 11	Feature 12
Decision Trees	0.0537	0.0233	0.0168	0.0170	0.0179	0.0016	0.0141	0.0153	0.0486	0.0593	0.0289	0.0164
Random Forest	0.0536	0.0295	0.0220	0.0272	0.0211	0.0056	0.0183	0.0233	0.0478	0.0584	0.0361	0.0223
LIME	0.0750	0.0372	0.0340	0.0448	0.0261	0.0154	0.0278	0.0369	0.0666	0.0770	0.0490	0.0371

Table 4.6: Mean Absolute Values of Feature Importance for Twelve Features

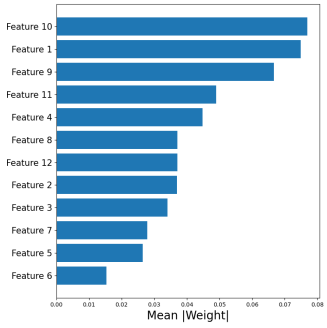
random forest, it is 0.0272, and for decision trees, it is 0.0179. Decision trees are further off in the mean absolute feature importance values with LIME than random forests. This can be attributed to the lower R2 scores of decision trees compared to random forests for higher feature settings. The feature importance of the three methods can be seen in Figure 4.7



(a) Decision tree feature importances



(b) Random Forest Feature Importances

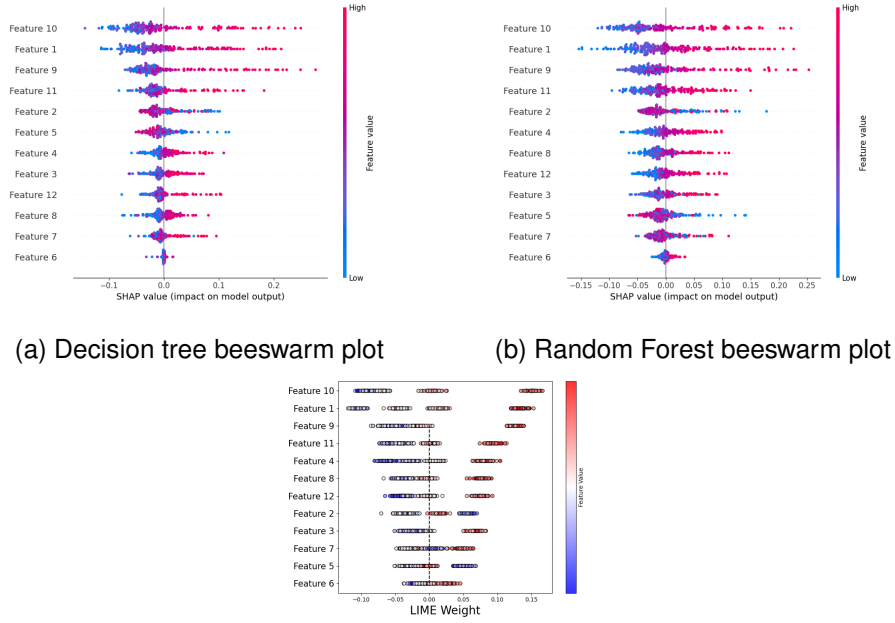


(c) LIME feature importances

Figure 4.7: Feature importances for 12 features and 4 participants per auction round

From the feature importance Figure, it can be seen, that the top four important features are the same in all three methods, even though surrogate models have a lower R2 score. Thereafter, there are some differences. The decision tree classifies Feature 5 as more important than Random Forest and LIME. This, however, is not significant as the features are very close in importance. The overall trend is similarly captured by all three methods, as can be seen in the beeswarm plot in Figure 4.8. For example, all three

convey that a higher value of Feature 10 corresponds to bidding a higher bid value.



(a) Decision tree beeswarm plot

(b) Random Forest beeswarm plot

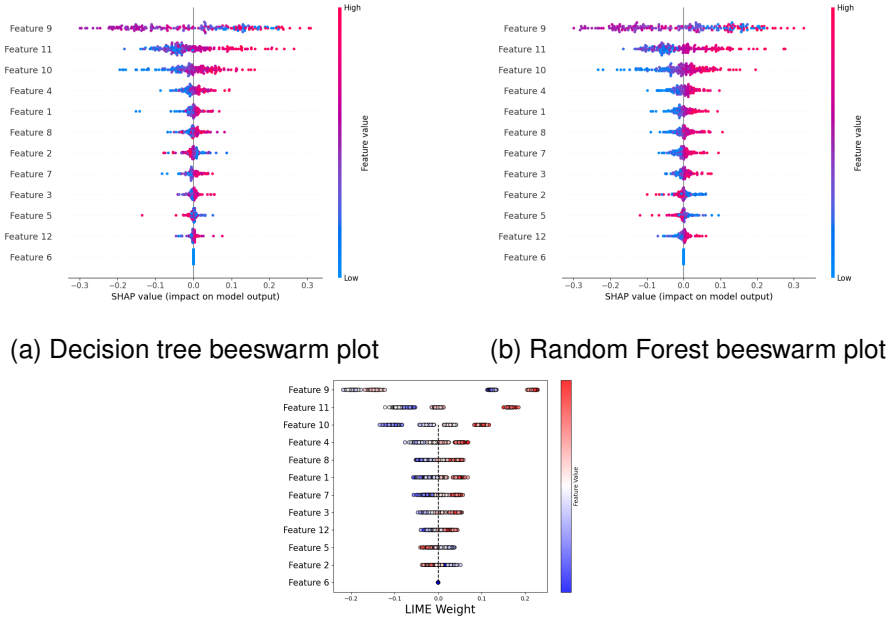
(c) LIME beeswarm plot

Figure 4.8: Beeswarm plot for 12 features with ground truth weight vector [1,1,1,1,1,1,1,1,1,1,1,1]

Subsequently, a ground truth weight vector of [1, 1, 1, 1, 1, 0, 1, 1, 4, 1, -3, 1] was employed. The underlying hypothesis was twofold: Feature 6, given its near-zero weight, should bear minimal importance, while Feature 9, with the highest weight of 4, should emerge as a key predictor of bid values. Moreover, due to its weight magnitude of 3 and negative coefficient, Feature 11 is anticipated to gain importance and reverse its outcome prediction in the case of linearity. The outcomes of the explanation methods are portrayed in Figure 4.9.

Remarkably, the magnitudes of feature importance in the results of all three explainable methods align with the anticipated ground truth. Notably, Feature 6's negligible weight corresponds to its minimal impact. Conversely, Feature 9's weight manifests its significance in bid prediction. In addition, the expected influence of Feature 11's negative weight magnitude is observed. However, an interesting observation arises: despite the negative weight, Feature 11's outcome prediction remains unaltered compared to its weight being 1. This phenomenon is illustrated in Figure 4.8. This result aligned with the results observed for the four feature setting and suggests the non-linearity of the auction gym model.

Similarly, like in the scenario involving four features, the context of 12 features



(a) Decision tree beeswarm plot

(b) Random Forest beeswarm plot

(c) LIME beeswarm plot

Figure 4.9: Beeswarm plot for 12 features with ground truth weight vector [1,1,1,1,1,0,1, 1,4,1,-3,1]

was subjected to perturbation. Each feature underwent scaling individually by a factor of 2, and the subsequent effect on the output bid was monitored. This evaluation was conducted by calculating the mean absolute change in the output bid value. The results of this analysis are presented visually in Figure 4.10.

The bar plot depicted in Figure 4.10 distinctly illustrates that through perturbing features and evaluating the entire test dataset, the top features highlighted by the explainable methods align with the fact that it caused more change in the output bid.

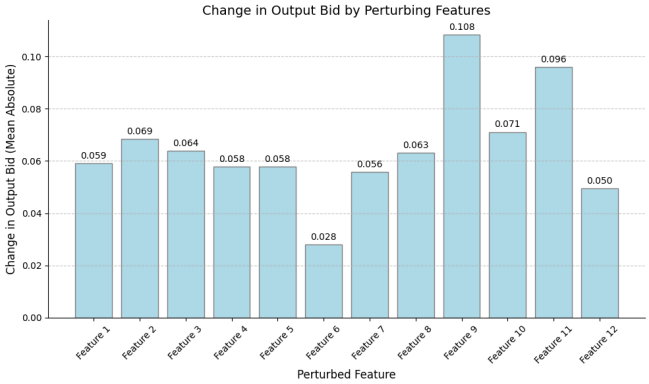


Figure 4.10: Effect in Output bid by perturbing for 12 features

Furthermore, the plot reinforces the anticipated result: Feature 6 is the least influential, aligning with the explainable methods.

Intriguingly, while there exists a slight variation in the ordering of features within the mid-range of importance as indicated by the explainable methods, the significance of this reordering is diminished due to its proximity among the features in terms of importance.

4.2.3 Statistical Test for Feature Importance across Explainable Methods

A statistical t-distribution test was conducted to ascertain whether the feature importances yielded by different explainable methods were statistically discernible across all features for two specific scenarios: one involving four features and the other involving 12 features with a ground truth weight vector as 1 for all features. The examination encompassed a dataset of 200 samples.

The results revealed that, for both scenarios, the feature importances derived from all methods were statistically indistinguishable. In other words, no significant statistical differences were observed among the feature importances obtained through various methods. This outcome was the same across all features and methods studied.

Specifically, the p-values associated with comparing feature importances were consistently greater than the predetermined significance threshold of 0.05. This suggests that the observed similarities in feature importance were not likely to have occurred due to chance. For a detailed breakdown of the p-values corresponding to feature importance across different methods, please refer to Appendix B.

4.2.4 Local Analysis with LIME

A local analysis with LIME of two data points for four features with ground truth weight vector of [1,-2,3,0] was analysed to see if the results remain consistent as conveyed by the mean absolute feature importance value from the 200 points. The first local data point is when the actual bid is low. LIME's decision on why it reached the conclusion of the bid value for this instance can be seen in Figure 4.11. It can be seen that Feature 3 is the most important in making the decision because of its high value, which corresponds to a lower bid. This can be related to the explanation provided by the 200 points in the beeswarm plot of LIME in Figure 4.5. The other three also remain consistent as per the beeswarm plot, i.e. a lower value of Feature 2 shifts the decision to make a lower bid, a

higher value of Feature 1 favours a lower bid, and Feature 4 does not affect the output at all.

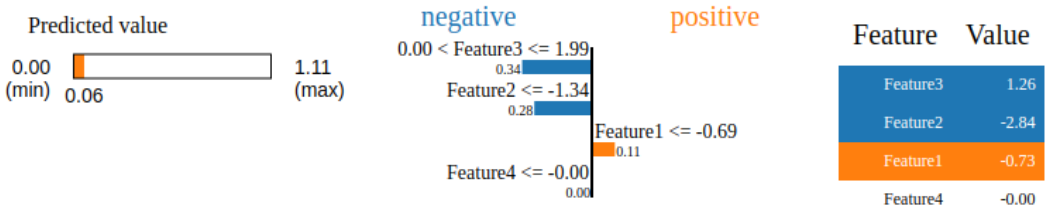


Figure 4.11: LIME explanation for low bid value

The second example is when the actual bid value is high. The explanation of this instance by LIME can be seen in Figure 4.12. For this instance, Feature 2 and Feature 1

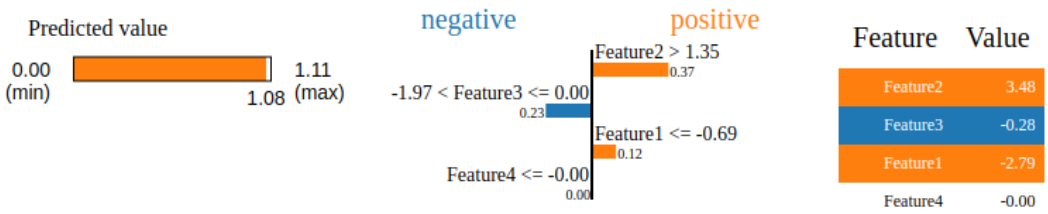


Figure 4.12: LIME explanation for high bid value

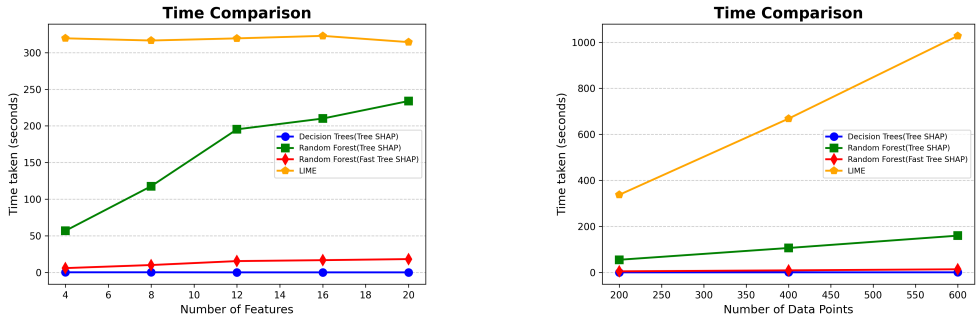
correspond to bidding a higher value, which is consistent with the beeswarm plot 4.5. However, Feature 3 favours a lower bid and is not the most important in deciding the bid value. For this instance, Feature 2 seems the most important in making this decision, which is slightly different from the mean feature importance of the four features where Feature 2 is the second most important. This is seen in the LIME explanation, where Feature 2 impacts the output more than Feature 3.

LIME provides a reason why the model reached a particular decision, which makes the model transparent. The trend seems consistent with the overall trend, with slight differences in the feature importance order for specific local instances. Thus, the explainability techniques help improve the interpretability of the doubly robust bidding estimator of the auction gym environment for online auctions.

4.3 Time Evaluation

Explaining the behaviour of complex models, such as neural networks and reinforcement learning agents, can be time-consuming. In contrast, less complex models like decision trees are generally faster to explain due to their simple and interpretable nature.

Experiments were conducted to evaluate the time taken to compute Shapley values on decision trees and random forests. Also, evaluation was performed on the time taken to compute the LIME feature importance on the auction gym doubly robust bidding agent. The goal was to compare the time taken to generate explanations for 200 data points for varying contextual features. It can be seen in Figure 4.13a the time taken to provide explanations by the different explainable methods with the increase of the number of features. As expected, LIME, performed on the complex Auction Gym RL agent itself, is significantly slower than decision trees as surrogate models. The complexity of the RL agent contributes to the additional computational overhead of LIME. On the other hand, decision trees offer much faster explanation times, making them more efficient for interpretability tasks. However, the decision trees suffer from poor performance as their R2 scores are less when the contextual features are high.



(a) Time taken for number of participants as 4

(b) Time taken for 4 features

Figure 4.13: Time Analysis for various Explainable Methods

An experiment was also performed to see the effect of explanation time on increasing the number of data points. Data points of 200, 400, and 600 were considered. From 4.13b, it can be seen that all explanation time of all the explainable methods increased linearly with the increase in the number of data points to explain.

To improve the accuracy of the surrogate model, random forest was experimented as a more sophisticated alternative to decision trees. Random forest typically yields better accuracy by ensembling multiple decision trees. However, the improved accuracy comes at the cost of increased computation time, as seen in Figure 4.13. The tradeoff between accuracy and explainability becomes apparent, where the random forest's higher accuracy is accompanied by slower explanation times. Decision trees provide a quick and interpretable solution but may not precisely mimic the original RL agent's behaviour in complex scenarios. On the other hand, random forest may offer better accuracy but at the expense of longer explanation times.

However, Fast Tree SHAP [19] applied on Random Forest increased its explanation speed significantly, as seen from the 4.13. It can be seen from Table 4.7 that the time of explanation for decision trees remains more or less the same with the increase of features. The time of explanation for Random Forest increases with the increase in the number of features. Fast Tree SHAP provides an alternative and faster way to provide explanations than regular Tree SHAP. For 20 features, Tree SHAP takes 233.9 seconds to compute the Shapley values for Random Forest, while Fast Tree SHAP is able to compute that in only 18.15 seconds. Also, the time to explain by LIME remains constant with the increase in the features. This is because LIME approximates a simpler model in the local region to explain a complex model. Thus, it does not significantly increase the time of explanation with a slight increase in the features. However, a very high number of features will start increasing the explanation time of Tree SHAP on decision trees and LIME on the doubly robust bidding estimator.

Technique	**num_features**				
	4	**8**	**12**	**16**	**20**
Decision Trees (Tree SHAP)	0.12	0.14	0.04	0.03	0.03
Random Forest (Tree SHAP)	56.77	117.65	195.28	210.01	233.90
Random Forest (Fast Tree SHAP)	5.84	10.04	15.42	16.68	18.15
LIME	319.54	316.52	319.46	322.87	314.34

Table 4.7: Time taken (seconds) for an explanation for 4 participants per auction round

Overall, it can be seen that surrogate models can provide a faster and more interpretable explanation for low features and less competitive auction settings. They can provide a global explanation of a high number of data points at a much faster pace than LIME.

Chapter 5

Conclusions

In this chapter, we will summarise the findings of our research on the explainability techniques in the study for explaining the behaviour of the doubly robust bidding agent in the auction gym environment. It will also be followed by a discussion on the limitations and the future work planned to be done.

5.1 Findings

The thesis researched the exploration of Explainable AI (XAI) techniques in the context of auction using the Auction gym simulation environment. The project aimed to interpret the novel Doubly Robust estimator bidding strategy proposed in the auction gym [6]. The aim was to provide a more interpretable understanding of the complex online bidding process and to increase trust in the decision-making process of the doubly robust bidding agent across various auction scenarios.

To achieve this, various experiments were conducted by varying the number of features and the number of participants in an auction round to explain the doubly robust estimator. Techniques of surrogate models of decision trees and random forests were employed to mimic the doubly robust bidding agent to provide a global explanation of the bidding agent. Additionally, TreeSHAP was used on the surrogate models to evaluate the feature importances for 200 data points. Meanwhile, LIME was used on the complex doubly robust estimator to find the feature importances for these 200 data points.

The feature importance of the explanation methods of Tree SHAP on surrogate tree models and LIME on the auction gym doubly robust estimator were compared with each other to see if they were similar in their explanations. Moreover, the feature importance

obtained by the various explainability methods was checked for consistency with the established ground truth. Furthermore, the time taken by the various explanation methods was examined.

5.1.1 Tree Surrogate Models

Random forests and decision trees were observed to emulate the auction gym model closely, achieving high R2 scores, particularly with fewer features and less competitive environments. However, introducing more features complicated the bidding behaviour of the doubly robust bidding estimator, making it challenging for tree-based surrogate models to replicate, thus lowering R2 scores. Notably, with a limited number of contextual features, R2 scores decreased as auction participants increased. Yet, as the quantity of these features expanded, this decline became less pronounced, with R2 scores even seeing slight improvements with more participants.

Tree-based models exhibited consistent performances across various agents for identical auction setups in multi-agent bidding settings. An in-depth R2 score comparison across diverse configurations highlighted that the model's performance remained largely unaffected by an increase in the number of agents, given that other factors stayed consistent.

5.1.2 Explanation by TreeSHAP and LIME

In our analysis involving auction settings with both 4 and 12 features, the explanations provided by Tree SHAP for decision trees and random forests exhibited similarities to those from LIME. However, the behaviour of the auction gym model proved challenging to capture accurately due to its inherent complexity and non-linearity. While all three explanation techniques accurately captured variations in the magnitude of a feature's ground truth, they faced limitations in reflecting changes in the direction of the bid outcome when the sign of a feature's ground truth vector was altered. This observation underscores the unique intricacies of the auction gym model, where a simple reversal of a feature's sign might not be sufficient to influence the bid's outcome direction predictably.

5.1.3 Speed of Explainabilty

In our studies, Tree SHAP demonstrated notable efficiency when explaining decision trees and random forests. Specifically, explanations for decision trees were computed in a mere 0.12 seconds, while random forests required a longer 56.77 seconds. In contrast, LIME took significantly longer—319.54 seconds—to explain the auction gym model with just four features.

It was observed that the explanation time for random forests using Tree SHAP increased with the addition of more contextual features. Yet, decision trees maintained their rapid computation speed regardless of the number of contextual features introduced, although their R2 score was lower than random forests. However, using Fast Tree SHAP [19] significantly improved the computing of feature importance for Random Forest from 56.77 seconds to just 5.84 seconds for the four feature setting.

Interestingly, the computation time for LIME remained consistent as the number of features increased. This consistency can be attributed to LIME's reliance on simpler models, such as linear regression, to approximate local regions. However, it is worth noting that a substantial feature increase might lead to prolonged explanation times. Additionally, across all the explanation techniques, a linear relationship was identified between the number of data points and the time taken for explanations: as data points increased, so did the computation time.

5.1.4 Comparative Analysis

In conclusion, given the impressive R2 scores achieved by tree-based models in emulating the Auction Gym's doubly robust estimator and their rapid explanation capabilities for scenarios with fewer competitors and contextual features, these models are valuable techniques for explanation. However, as the environment's complexity escalates, decision trees face challenges in accurately mirroring the model, reflected in diminished R2 scores.

Tree SHAP and LIME can also capture the feature importance following the magnitude of the linear ground truth weight vector. However, the effect of change in sign in ground truth vector for a feature was ineffective in changing the direction of bid outcome for that feature. This suggests that the behaviour of the auction gym is intricate and more complex than a linear model. The surrogate models explanation by Tree SHAP outpaces LIME in speed. While Random Forest's explanation time via Tree SHAP escalates with more features, introducing Fast Tree drastically reduces this duration.

5.2 Limitations

The experiments have been run for a maximum of 20 features and a maximum number of participants in an auction round of 6. Also, a fixed advertisement inventory of 12 has been considered in the experiments. Companies may have higher advertisements to show in real-world online auctions. The experiments are not on real-world data and on simulation data of the auction gym. Grid Search is not performed on random forests due to its time complexity, and there may be a slightly more optimal tree for scenarios with a higher number of contextual features.

5.3 Future Work

For our further work, we would like to consider working with a higher number of observed features and also working with a varied number of advertisements in the inventory of the bidding agents in the auction gym environment. Further, we would like to work with a higher number of data points for an explanation for LIME and Shapley values to find a more global interpretation of the feature importances. Additionally, we would like to establish different ground truths like quadratic and see if the explainable methods can capture that pattern, giving more insights into the auction gym model. In addition to this, we would consider the performance of the explainability of surrogate models on the Second Price Auction and the Oracle Allocator for complex settings and compare it with the results of the First Price Auction.

Bibliography

[1] Amina Adadi and Mohammed Berrada. Peeking inside the black-box: A survey on explainable artificial intelligence (xai). *IEEE Access*, 6:52138–52160, 9 2018.

[2] Shahin Atakishiyev, Mohammad Salameh, Hengshuai Yao, and Randy Goebel. Explainable artificial intelligence for autonomous driving: A comprehensive overview and field guide for future research directions. 12 2021.

[3] Moritz Böhle, Fabian Eitel, Martin Weygandt, and Kerstin Ritter. Layer-wise relevance propagation for explaining deep neural network decisions in mri-based alzheimer's disease classification. *Frontiers in Aging Neuroscience*, 10, 2019.

[4] Siu Lun Chau, Robert Hu, Javier Gonzalez, and Dino Sejdinovic. Rkhs-shap: Shapley values for kernel methods. 10 2021.

[5] Prashant Gohel, Priyanka Singh, and Manoranjan Mohanty. Explainable ai: current status and future directions. 2021.

[6] Olivier Jeunen, Sean Murphy, and Ben Allison. Learning to bid with auctiongym, 2022.

[7] Scott Lundberg and Su-In Lee. A unified approach to interpreting model predictions. 5 2017.

[8] Scott M. Lundberg, Gabriel G. Erion, and Su-In Lee. Consistent individualized feature attribution for tree ensembles. 2 2018.

[9] John A. McDermid, Yan Jia, Zoe Porter, and Ibrahim Habli. Artificial intelligence explainability: The technical and ethical dimensions, 10 2021.

[10] Andreas Messalas, Yiannis Kanellopoulos, and Christos Makris. Model-agnostic interpretability with shapley values. 2019.

[11] Jean Jacques Ohana, Steve Ohana, Eric Benhamou, David Saltiel, and Beatrice Guez. Explainable ai (xai) models applied to the multi-agent environment of financial markets. volume 12688 LNAI, pages 189–207. Springer Science and Business Media Deutschland GmbH, 2021.

[12] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. 2 2016.

[13] Lloyd S Shapley. A value for n-person games. *Contributions to the Theory of Games 2.28*, pages 307–317, 1953.

[14] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences, 2017.

[15] Alexander Sieusahai and Matthew Guzdial. Explaining deep reinforcement learning agents in the atari domain through a surrogate model, 2021.

[16] Timo Speith. A review of taxonomies of explainable artificial intelligence (xai) methods. pages 2239–2250. Association for Computing Machinery, 6 2022.

[17] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks, 2017.

[18] W. Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of Finance 16, 1*, pages 8–37, 1961.

[19] Jilei Yang. Fast treeshap: Accelerating shap value computation for trees. 9 2021.

Appendix A

Correlation Matrix for Contextual Features

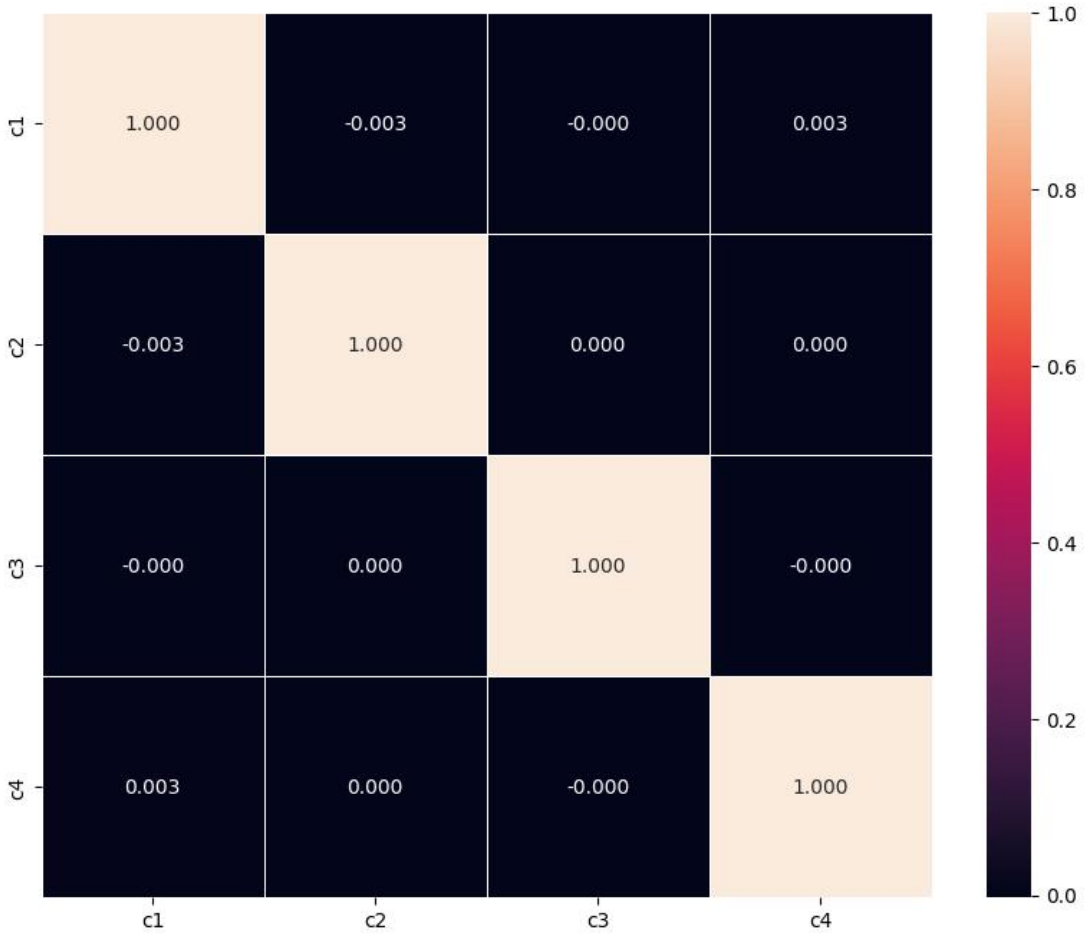


Figure A.1: Correlation Matrix for 4 context features for agent 0 with number of participants in an auction as 4

Figure A.2: Correlation Matrix for 12 context features for agent 0 with number of participants in an auction as 4

Appendix B

Statistical Significance t-distribution test between Feature Importances

The significance value for the t-distribution test is set as $\alpha = 0.05$.

Methods	**Feature Importances**			
	Feature 1	**Feature 2**	**Feature 3**	**Feature 4**
Decision Trees & Random Forest	0.995	0.968	0.913	0.904
Decision Trees & LIME	0.762	0.894	0.704	0.884
Random Forest & LIME	0.757	0.924	0.626	0.804

Table B.1: Statistical Test p-values for 4 features between different methods (200 samples)

Methods	**Feature Importances**											
	Feature 1	**Feature 2**	**Feature 3**	**Feature 4**	**Feature 5**	**Feature 6**	**Feature 7**	**Feature 8**	**Feature 9**	**Feature 10**	**Feature 11**	**Feature 12**
Decision Trees & Random Forest	0.935	0.883	0.661	0.775	0.924	0.640	0.485	0.896	0.946	0.882	0.858	0.636
Decision Trees & LIME	0.687	0.806	0.945	0.934	0.431	0.998	0.162	0.460	0.845	0.882	0.936	0.572
Random Forest & LIME	0.741	0.925	0.798	0.909	0.418	0.836	0.450	0.559	0.795	0.985	0.822	0.843

Table B.2: Statistical Test p-values for 12 features between different methods (200 samples)