Summer 2024

Explainable Artificial Intelligence: Methods and Evaluation

Gayane Grigoryan
Old Dominion University, grigory.gayaneh@gmail.com

EXPLAINABLE ARTIFICIAL INTELLIGENCE: METHODS AND EVALUATION

by

Gayane Grigoryan
M.A. May 2017, Old Dominion University
B.A. May 2012, Armenian State University of Economics

A Dissertation Submitted to the Faculty of
Old Dominion University in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

ENGINEERING MANAGEMENT AND SYSTEMS ENGINEERING

OLD DOMINION UNIVERSITY
August 2024

Approved by:

Andrew J. Collins (Director)

Steve Cotter (Member)

David Selover (Member)

ABSTRACT

EXPLAINABLE ARTIFICIAL INTELLIGENCE: METHODS AND EVALUATION

Gayane Grigoryan
Old Dominion University, 2024
Director: Dr. Andrew J. Collins

A wide array of techniques within explainable artificial intelligence (XAI) have been developed to measure the importance of features in machine learning models. A notable portion of these methods draws upon principles of cooperative game theory (CGT), with the Shapley value emerging as a widely used solution concept. Despite the rising prominence of the Shapley value, other promising solutions from cooperative game theory—such as the Nucleolus, Banzhaf power index, Shapley-Shubik power index, and solutions to conflicting claims problems—have been comparatively overlooked, even though they hold significant potential. In this dissertation, multiple XAI methods based on these other CGT solutions are proposed. These methods were applied in both linear and classification scenarios, addressing datasets with both independent features and multicollinearity concerns. Prior work considered the sensitivity of explanations through permutation tests or the accuracy of explanations to evaluate XAI methods. However, these approaches do not address the uncertainty or the consistency associated with the feature importance evaluations. In this dissertation, a weighted Shannon entropy-based permutation relative importance evaluation (PRIME) metric is proposed to assess the consistency of feature importance methods in determining the relevance of the features. This metric integrates the established methods of permutation tests and weighted Shannon entropy to conduct the evaluation. The novelty of this dissertation lies in (1) demonstrating the applicability of numerous CGT solutions to measure feature importance values, (2) showing the effectiveness of these techniques using permutation relative importance evaluation metric, and (3) employing these methods to investigate input data that can be used for an agent-based model. The results show that the Shapley-Shubik, Banzhaf power index and conflicting claims problems-based feature importance methods offer advantages over Shapley value-based methods due to their unique properties when explaining feature importance values. The findings also demonstrate that PRIME can effectively evaluate feature importance methods.

Dedicated to my family.
To my mother - Anush Khanzadyan-Grigoryan, who has always prayed for me.
I love you all.

ACKNOWLEDGMENTS

My Ph.D. journey has been an unforgettable experience filled with achievements, joy, and constant blessings. I am grateful to Jesus for His blessings and love no matter where I find myself or what tasks I undertake. God has blessed me with wonderful people who have brought joy and happiness into my life.

I would like to thank my advisor, Dr. Andrew Collins, for his constant support and wisdom in making my Ph.D. experience productive. I am grateful for his encouragement during times of doubt. I will always cherish his advice, and I am thankful for the moments we shared discussing research, publications, life, and the journey beyond Ph.D. I was always confident that any questions I had would be addressed.

I am grateful to my committee members, Dr. Steven Cotter and Dr. David Selover, for the insightful discussions and suggestions to improve my work. Whenever I was homesick, I would visit Dr. David Selover's office, where he would welcome me with stories about Armenia, bringing comfort and a sense of home. I appreciate how patient Dr. Steven Cotter is and how he would explain any statistical method so thoroughly until everything is completely clear.

I want to say thanks to everyone in my department. I appreciate the discussions and ideas that Ariel Pinto shared with me until the last stages of my dissertation and working on the Spencer project. I am grateful that I worked with Sam Kovacic, and I value his constant support, care, and encouragement. I am grateful for the discussion about Hawaii and all the research meetings we had, which were encouraging and motivating to find new ways to explore the research questions.

Working alongside James Leathrum, Ross, Chris, Sol, and Jannette on the NAVSEA project was an incredibly rewarding experience. VMASC offered an enriching experience filled with learning, challenging me to approach research from a critical standpoint, constantly questioning and seeking to enhance every aspect, and I am grateful for this experi-

ence with Jose, Hamdi, and Daniele. I will always cherish my MSVE friends, my dear sister Shrabani, and friends Ahmet, Haben, and Andy. I cherished our Friday dinners, gym sessions, hikes, and the weekend walks we would take together. My Engineering Management (ENMA) friends Ying, Wael, Ikram, Farinaz, Ahmed, Sheida, Sujata, and Francisco, I am grateful to you all.

Norfolk, VA, is my second home because of all the wonderful people I met here. I am extremely grateful to the Norfolk family, Elizabeth, Candy, David, and Grandma Betsy, for their constant love and care. The day I arrived at their home was truly a blessing. I loved going to church, the delicious lunches, and the time we would spend together. Thank you for accepting me into your family and showing so much love and care. I am grateful to Cindy and Uncle Sam for always being with me and for their prayers. I am thankful to Aunt Natalie and Uncle Ross for the time we spent together. They always showered me with unwavering love, care, attention, and lots of homemade food. I am grateful for the love of sweet Mama Mary, for her kindness and strength to shower everyone around her with love and blessings. Many thanks to Barbara, who helped me create my first CV and encouraged me to apply to study abroad. She introduced me to Fulbright and spent hours working on the application and cover letter. I am grateful for my sweet sisters, Aida and Hannah. You all brought joy to my life: Mari, Hasmik, Anzhelika, Diane, Faith, Allison, Adam, Eveline, Najmin, Mahfoudah, Siranush, Mariam, and many others.

There are no words to express how grateful I am to my family. I am blessed with the most supportive family. My parents, Anush and Arsen Grigoryan, my fiance Gevork and the family Noune and Aroutioun, my brother Zhora, my grandma Fenya, my aunt Lusine, and grandma Greta, I love you all so much, and I know that you all are a blessing from God. Gevork, thank you for your love, kindness, and strength. I treasure every moment we spend together and look forward to all that awaits us in the future. My mom has always been a role model for me with her hard work, patience, positivity, dedication to her faith, and caring nature towards our family. Mom, you are the most precious and adorable blessing God has bestowed upon me. I am grateful to my dad and my brother for their love and support to achieve my dreams and goals. My grandma is the cutest. She is strong, hardworking, kind, and very determined to achieve any goal she sets her mind to. Despite the geographical distance that has separated us, their hearts are always close to mine, echoing any challenges I am facing and rejoicing in my happiness. My parents gave me every opportunity to succeed in life and made Zhora and me their priority. I have more family and friends whom I love dearly and who have immensely contributed to my personal and professional growth; I want to say thanks to everyone.

I thank God for all these blessings and all these people in my life. I love you all.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

This chapter presents the problem statement, objectives, research questions, contributions, the impact of the study, the publications, and the dissertation overview.

Machine learning (ML) techniques have become pervasive in the last decades. Machine learning models refer to computational systems designed to learn patterns and make predictions or decisions based on data [2]. These models are trained on a dataset, using various algorithms to capture underlying patterns and relationships within the data. Different features (input variables) contribute to the model's overall performance, and assessing the importance of these features becomes crucial for understanding model behavior [3].

However, grasping the behavior of the models — how they operate and explain the rationale behind specific decisions— can be challenging. [4]. Explainability is the ability to describe what happens in the model from input to output [5]. Explainable artificial intelligence (XAI) is a more formal way to describe this within the artificial intelligence domain. XAI helps to make the model transparent and addresses the black box problem [5].

The need for explainability is primarily motivated by the social dimension of explanations [6]: improve trust towards the model-supplied outcome and avoid legal or ethical issues and potential biases and discrimination. For example, XAI is necessary for reinforcement learning as these models are considered black boxes due to the lack of transparency in understanding the model's inner workings and decision-making process [7]. Simple, explainable models, such as a regression model, can also be challenging to interpret. These models do not automatically guarantee explainability due to issues such as multicollinearity. Multicollinearity can result in different coefficients with equivalent accuracies. To avoid this issue, the number of input features could be limited during model training using some regularization, such as Lasso [8], Ridge regression [9], or perform principle component analysis

[10]. However, these techniques are limited in data interpretation [11, 12]. Instead, the use of explainable artificial intelligence has been advocated to enhance the model's explainability [13, 14].

XAI provides techniques to improve the explainability of AI decisions and predictions. XAI describes an AI model, its expected impact, and potential biases. It helps characterize model accuracy, fairness, transparency regarding how the model reached a particular answer, justification of why the output should be accepted, and the information to back decisions. Numerous XAI techniques exist [15, 13, 14, 16, 17, 18, 19]. From these techniques, a subset of techniques known as feature importance, also known as feature attribution or credit assignment techniques based on cooperative game theory, will be analyzed in this dissertation.

In more sophisticated machine learning models, hundreds of features could be used to realistically describe the real-world system. Explaining which of these features is more important for the system representation and functionality is a significant aspect we will address in the scope of this dissertation. Feature importance (FI), also called feature attribution [20] and credit valuation [21], measures the contribution of the input variables, or features, to the ML model functionality and its performance [22]. To determine the feature importance values, the model performance, such as the increase of model error, is computed after removing a particular feature. If removing the feature from the model does not affect the model performance, then the feature is not important for the model. Feature importance is part of the post-hoc explanation generation [23]. Post hoc explanations mean applying an explainable model, such as a regression model, to a target model to extract explanations and describe the predeveloped models [23]. Post-hoc explanations can be treated as proxy or surrogate models to the original model.

There are many feature importance methods due to the benefits these methods introduce for selecting essential features and model explanations. Shapley value, a cooperative game-theory-based solution, has been prevalent in developing feature importance methods [24]. The reason for this prominence is the robust mathematical framework Shapley values provide to explain the dynamics of a system. The objective of Shapley-value-based feature importance measures is to identify features' respective contributions to the model performance to improve the system's possible outcome. Generally, the cooperative game theory seeks to determine what coalitions will form and how to divide the payoff (rewards) among coalition members fairly [24]. In the model, features or agents are the coalition members, and the payoff could be the model performance. This dissertation focuses on other cooperative game theory solution concepts to measure feature importance values.

1.1 MOTIVATION/PROBLEM

This section outlines the motivation behind the dissertation and defines the problem it seeks to address.

Shapley value suffers from several limitations related to its axioms that do not generally guarantee that the Shapley value is suited to feature importance [25, 1]. Kumar et al., [26] describe mathematical and human-centric issues associated with Shapley-value-based explanations as feature importance measures. Studying other cooperative-game theory-based solutions could help address the challenges and limitations or provide new insights rather than relying solely on the Shapley value to measure feature importance in machine learning models. Other cooperative-game theory-based solutions could provide alternative ways of modeling the interactions and dependencies among features and help identify potential biases or limitations of machine learning models that may be missed by Shapley value. Often, these solutions, such as Nucleolus, Shapley-Shubik, or Banzhaf power index, are chosen because they may lead to various unique predictions and provide different explanations about the problem of interest. For example, the core evaluates which coalitions will form by determining the feasible payoff allocations that another subset of players cannot improve upon [27]. The nucleolus of a model is a solution concept that minimizes the worst inequity of the coalition [28]. Shapley-Shubik determines how likely it is for a feature to become pivotal [29]. Banzhaf power index would be a numerical representation of how likely the feature is to be critical and substantially influence the final outcome [30]. Generally, solution concepts declare rules for predicting how a game will be played, and these predictions are considered solutions. The essence of these solution concepts is fairness and rationality. To determine which solution concept is most appropriate to measure feature importance values for a particular prediction, I experiment with different solution concepts and see which one provides the most human-centric results. Human-centric refers to a focus on machine learning outcomes tailored to be understandable, relevant, transparent, and beneficial from a human perspective, especially in terms of interpretability, responsibility, and usability [31]. This involves designing and selecting solution concepts that produce results that humans can easily interpret, relate to, and apply in decision-making processes, ensuring that artificial intelligent system serves to augment human capabilities and knowledge [32].

In essence, this dissertation aims to assess the applicability of various cooperative game theory solutions to determine feature importance values using linear regression, logistic regression, and input data that could be used to enhance agent-based models. A linear regression model is a predictive model that describes the relationship between one or more features and the target variable [33]. Logistic regression is a popular algorithm used for binary classification, where the task involves predicting one of two classes [34, 35]. Agent-based models are simulation models that replicate the behavior of individual entities, known as agents, within a defined environment [36]. These models are designed to capture the dynamic interactions and decision-making processes of agents, often reflecting the complexity and emergence observed in real-world systems.

1.2 AIM AND OBJECTIVES

In this section, the objective of the dissertation is presented.

The objectives of this research are to develop new cooperative game theory-based explainable AI (CGTXAI) methods, apply these new methods to linear and logistic regressions and agent-based models, and consider the Weighted Shannon entropy permutation-based evaluation approach (PRIME) to assess these methods' performances. The objectives of this dissertation are listed below:

1. *Application of cooperative game theory-based XAI methods to the linear model.* Regression models are considered inherently explainable; however, it was observed that the regression model explainability reduces when multicollinearity is present. Shapley values have been applied to explain the feature importance values in regression models with multicollinearity issues [37]. However, other cooperative game theory solutions have not been used to evaluate the regression feature importance values. In the scope of this dissertation, several cooperative game theory solution concepts will be considered to evaluate the regression feature importance values.

2. *Application of cooperative game theory-based XAI methods to the classification model.* Here, logistic regression is used as a surrogate model to explain more complex black-box models [13]. Surrogate models are a concept in machine learning used to interpret and explain the decisions made by more complex models—commonly referred to as black box models [13]. Developing explainable AI techniques tailored for logistic regression is crucial, particularly when faced with the challenge of elucidating the decision-making processes involved in more intricate classification tasks.

3. *Application of cooperative game theory-based XAI methods to study input data that could be used when designing agent-based models.* Agent-based models are used to study complex systems where individual agents, each characterized by unique attributes, governed by a set of rules and behaviors, interact within an environment to simulate emergent phenomena [36, 38]. Agent-based models are known for their sensitivity to input and parameter changes [39, 40]. Considering cooperative game theory-based explainable AI methods to explain the impact of parameter and input data variations in agent-based models holds significant potential for providing a comprehensive understanding of the convoluted dynamics governing these systems.

4. After exploring new explainable AI methods across diverse models, this dissertation aims to provide a quantitative way to assess the performance of cooperative game theory-based explainable artificial intelligence methods. Weighted Shannon entropy-based permutation relative importance evaluation (PRIME) is developed to measure the consistency of feature importance ranking. This allows for a systematic and objective evaluation, enabling comparisons between different methods and determining which ones are more effective in providing meaningful explanations.

1.3 RESEARCH QUESTIONS

In this section, the research questions are outlined.

The main hypothesis is that various new cooperative game theory-based explainable artificial intelligence techniques can be pertinent ways to explain feature importance values for regression, logistic, and agent-based models. The following research questions will be addressed in the scope of this dissertation:

1. To what extent are various cooperative game theory-based solutions (Shapley values, core, Nucleolus, Shapley-Shubik, Banzhaf power index, and conflicting claims problem) useful to explain the feature importance values for a regression model with multicollinearity issue?

2. To what extent various cooperative game theory-based solutions (Shapley values, core, Nucleolus, Shapley-Shubik, Banzhaf power index, and conflicting claims problem) can explain a logistic regression model with independent features?

3. To what extent various cooperative game theory-based solutions (core, Nucleolus, Shapley-Shubik, Banzhaf power index, and conflicting claims problem) can explain the changes in input data suitable for integration into agent-based models?

4. To what extent weighted Shanon entropy-based permutation importance evaluation

(PRIME) can measure the performance of methods across diverse models, considering factors such as consistency and uncertainty?

These questions aim to enhance the feature importance measures, provide additional information to the target model, and improve the trust and understanding of the model outcomes for stakeholders interested in the prediction.

1.4 RESEARCH CONTRIBUTIONS AND OUTCOMES

The research contributions are as follows:

- This is the first attempt to apply various cooperative game theory solutions, such as Nucleolus, Shapley-Shubik power index, Banzhaf power index, and conflicting claims solutions (conflicting claims proportional, constrained equal awards, constrained equal losses, Talmud and constrained random arrival) to advance model explainability and feature importance measures using linear and logistic regression models. The results show that other solutions are as important as Shapley's value, which has been usually used.
- This is the first attempt to apply various cooperative game theory solutions, such as Shapley values, to advance model explainability and feature importance measures to study input data for agent-based models.
- Weighted Shannon entropy-based permutation relative importance evaluation metric (PRIME) is the first attempt to measure the feature importance methods' consistency when evaluating the influences of the features on the prediction.

1.4.1 RESEARCH IMPACT

Nowadays, machine learning algorithms analyze user data and affect the decision-making process in areas like medicine [41], employment [42], traffic control [43], education

[44] and criminal justice [45]. These algorithms could have biases and result in discrimination and unfair decision-making [46, 47]. Even machine learning experts may have difficulty fully comprehending the inner workings of the algorithm, and this complexity can hinder debugging and, thereby, impeding their adoption [48]. However, it is vital to provide means for communicating the algorithm's findings with non-experts. Explainable Artificial Intelligence (XAI) methods can be a solution to obtain a more complete understanding of the prediction [15, 14]. New cooperative game theory explainable artificial intelligence methods can be instrumental in enhancing the overall model understanding and transparency while enabling the identification of important features.

1.4.2 PUBLICATIONS

Here are the published and work-in-progress papers developed in the scope of this dissertation.

Published

- **Grigoryan, G.,** and Collins, A. J. (2023, December). Feature Importance for Uncertainty Quantification In Agent-Based Modeling. In 2023 Winter Simulation Conference (WSC) (pp. 233-242). IEEE.

- **Grigoryan, G.,** and Collins, A. J. (2021). Game theory for systems engineering: a survey. International Journal of System of Systems Engineering, 11(2), 121-158.

- **Grigoryan, G.** (2022, June). Explainable Artificial Intelligence: Requirements for Explainability. In Proceedings of the 2022 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation (pp. 27-28). (Extended Abstract)

- **Grigoryan, G.,** Robaldo, L., Pinto, A., and Collins, A. J. (2023). Exploring the explainability and legal implications of regression models in transportation domain. In Juris-informatics (JURISIN). Workshop publication.

- **Grigoryan, G.,** and Collins, A. J. (2022). Is explainability always necessary? Discussion on explainable AI. In Modeling, Simulation and Visualization student capstone conference. Norfolk, VA.

1.5 DISSERTATION OVERVIEW

The outline of this proposal is as follows. Chapter 2 describes the background of explainable artificial intelligence, its methods, and the need for explainability. Chapter 3 discusses the research methodology. Chapter 4 presents the results. Chapter 5 concludes the dissertation.

CHAPTER 2

RELATED WORK

The goal of this chapter is to provide the essential concepts on explainable artificial intelligence (XAI), covering methods, game theory, various models including linear and logistic regression as well as agent-based models, and delving into the principles of systems engineering.

Usually, people do not feel comfortable agreeing with a machine learning (ML) system's decision without a complete understanding of the decision-making rationale of the system [49]. Extensive explanations of the machine learning model's output may be necessary to achieve its full credibility. An analyst or a machine learning expert may have good knowledge about the inner workings of the algorithm. However, it is vital to communicate algorithm findings with non-experts clearly by providing more information about the relationships between the features. This includes providing a transparent explanation about how the model reached a particular solution and a justification of why one should accept that result [50]. The models that are hard to comprehend are usually described as "black box" models, referring to an increased level of uncertainty in understanding the algorithm outcomes [51]. Simpler machine learning models, also known as white box models, can be easily understood by humans due to their lack of rules that design the model and generate the outcome [51]. An example of a less complicated white box machine learning model is the regression model, and an example of a black box model is a convolutional neural network. Many analysts blindly 'accept' the outcome of the black-box model, whether by necessity or by choice [52], without fully understanding why certain decisions were made. A black box model in machine learning refers to a type of computational model that makes decisions or predictions based on input data without revealing its internal decision-making process [16].

An example of an incorrect prediction of a machine learning algorithm is discussed by Ribeiro, Singh, and Guestrin [13], who have conducted experiments to distinguish photos of wolves and huskies (Figure 1). The machine learning model used was a logistic regression on a set of 20 images, hand-selected such that all pictures of wolves had snow in the background, while pictures of huskies did not. The classifier predicted "wolf" if there is snow (or a light background at the bottom) and "husky" otherwise. This means the classification model inadvertently learned to use the presence of snow in the background as a primary feature for prediction rather than more relevant features related to the animals themselves. If snow was in the image, the model predicted wolf; if not, it predicted husky. This reliance on an irrelevant feature highlights the potential pitfalls of machine learning models that might not make decisions based on the actual relevant features. Ribeiro, Singh, and Guestrin [13] have further conducted human subject experiments to show the classification model with and without explanation and emphasize the role of explanation in detecting incorrect prediction of the model and improving the trust towards the model-supplied outcome.

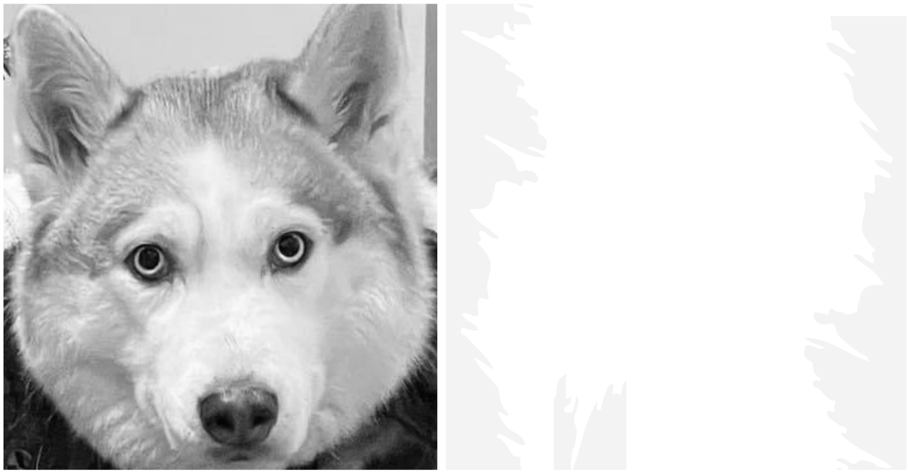


FIG. 1: Husky vs. Wolf experiment, showing a prediction of a husky as a wolf when the background has snow (in the left), and the snow (in gray) as the most important feature (in the right)

The results of this experiment show that the machine learning model prioritizes specific features over others—such as animal color, position, and facial structure — deeming them less important for classification. This experiment aimed to underscore the significant impact of the most important feature, namely snow, on the prediction outcome, using the LIME approach for distinguishing between huskies and wolves. As shown in Figure 1, with incorrect classification, the necessity to have a better understanding of the machine learning modeling decisions becomes imperative. This experiment showcases a critical aspect of machine learning models — the importance of explainable and trustworthy predictions. The experiment was designed to address the challenge of understanding and trusting the predictions made by machine learning algorithms, especially when these algorithms act as black boxes that are hard to interpret.

Explainable artificial intelligence (XAI), a subdomain of artificial intelligence, focuses on solving black-box-related issues when explanations are crucial [15, 22]. The following subsections elaborate on explainable artificial intelligence and the explainable artificial intelligence methods.

2.1 EXPLAINABLE ARTIFICIAL INTELLIGENCE

This section delves into the fundamental principles of explainable artificial intelligence.

Machine learning is an evolving branch of computational algorithms that are designed to emulate human intelligence by learning from the surrounding environment [53, 54, 55]. Machine learning algorithms include our opinions embedded in code and sometimes reflect human biases that lead to machine learning mistakes [56]. Explainable artificial intelligence (XAI), sometimes also referred to as explainable machine learning (XML) interpretable AI or interpretable machine learning (IML) has been extensively used to describe an ML model, its expected impact, and potential biases in the model's decision-making reasoning [57, 58, 59]. Explainability is generally described as the ability of the human user to understand the model's logic. Gregor and Benbasat [60] describe explainability as a declaration of the meaning of words spoken and actions to adjust a misunderstanding or reconcile differences. Explanations help to understand the system's malfunctions or anomalies [61]. The explanation is assumed to be provided by some source of information and that the explanation is geared to supply some data, knowledge, and evidence [50]. Gunning et al. [62] state that explainability reflects the objective to create more human-understandable ML models through the use of effective explanations. XAI provides answers to how and why particular decisions were made and if those decisions have been made for right or wrong reasons [63]. Rosenfeld and Richardson [64] state that the explainable system is the most central and essential for the systems' functionality. Gilpin et al. [65] argue that interpretability alone is insufficient.

The field of XAI is not new and can be traced back to the origins of artificial intelligence research and the development of expert systems [66]. Defence Advanced Research Projects Agency (DARPA) launched the Explainable AI program in 2017 [62] to emphasize the need for explainable models with high levels of learning performance (prediction accuracy); and to enable human users to understand, appropriately, trust, and effectively manage the emerging generation of artificially intelligent partners.

Three key stakeholder groups interested in XAI analysis are developers, end-users, and regulators [67]. The first group is the engineers that build the ML models. Engineers seek to gain a deeper understanding of the model and improve its performance. The second group is the consumers or end users who may not have the technical knowledge and skills to understand how the algorithm works. However, understanding the model outcome is vital for the end users as it builds trust that model decisions are reliable and equitable. The final group of stakeholders is the regulators who want to ensure model decisions comply with laws and do not amplify undesirable bias from the underlying model specification and the data. XAI can offer improved insights to regulators to trace unexpected predictions and identify corrective actions.

Several explainability-related terms exist, such as interpretability, transparency, fairness, explicitness, and faithfulness [68, 50]. XAI is a key part of broader human-centric responsible artificial intelligence practices. Interpretability focuses on model understanding techniques, while explainability focuses on model explanations and the interface for translating these explanations into human-understandable terms for different stakeholders. These methods bridge machine learning and human systems. For engineers, XAI is important for the accurate design of ML models based on the system's functionality.

For interpretability, the context and domain knowledge of the operator plays a crucial role [57]. An interpretable model example is how sales improve or decline as a direct result of the number of advertisements changing. Here is an example of explainability and interpretability. Think of an instructor that explains the material to the students. The instructor explains X, but a student may interpret the X as Y due to a lack of understanding of the topic; either the explanation was not clear enough, or the student does not know enough about the topic. For simplicity, I do not consider the case that the student may lack focus or interest in the subject. Interpretation of the topic could be evaluated in various ways, such as by getting feedback from the student. However, the goal of explainability is to deliver the topic using methods that are clear for the student. To summarize, we can say that explainability is generated following a sequence of interpretations of the content. These two concepts, interpretability, and explainability in machine learning, are associated with some level of risk due to a random algorithmic bias that may skew the result [57].

The biased predictions and misrepresentations could affect numerous aspects of our lives, including algorithms used for hiring, medical or judicial predictions, traffic incidents, and financial trading. For a specific example, consider a hiring algorithm that helps to navigate the Curriculum Vitae of potential job candidates. This model could be fallible and have discriminatory effects on hiring practices for women, ethnic minorities, and other legally protected groups [69]. This discrimination is not usually observed in the model outcome. The objective of these algorithms is to help reduce the time spent reading resumes that do not match job requirements. Instead of searching through resumes by hand to select candidates that meet certain professional requirements, hiring personnel can rely on an algorithm designed to filter candidate resumes automatically. To trust the selection results, the recruitment procedures based on these automatic algorithms should be clear and fair regardless of gender, race, ethnicity, disability, economic status, and other diverse backgrounds. This is the reason we want the ML model outcome to be explainable so that we understand that the candidate selection is predicted based on her professional ability rather than other factors.

Therefore, the legal and ethical challenges of explainable artificial practices raise significance for tackling discrimination and providing transparency in machine learning models [70]. In the legal aspect, clarifying why a particular outcome was achieved is vital to confirm that the hiring process was fair and that the selection or rejection was not based on bias [70]. Here, the legal issues are based on a set of rules and are punishable by law if those rules are not observed [71]. Ethical issues are governed by some standards but are not punishable by law [71]. Eliminating these challenges is essential to achieve fair decisions [70].

2.2 EXPLAINABLE ARTIFICIAL INTELLIGENCE METHODS

This section outlines the methods of explainable artificial intelligence.

Several explanation methods and strategies have been proposed to make AI systems explainable. This section provides an overview of these methods. Adadi and Berrada [22] classify XAI methods based on the following three criteria: (i) used model type or the level of dependency from the used ML model (ii) the scope and (iii) the complexity . Figure 2 shows the categories of XAI methods.

Below, I describe the methods included in Figure 2.

Based on the used model, explainability is model-specific and model-agnostic. Model-specific refers to the explainability techniques that are specifically designed for a particular modeling paradigm. Model-specific techniques are wide-ranged. For example, the Shapley net effects technique was developed by Lipovetsky and Conklin [37] discusses feature importance in the context of a multiple regression model. The Shapley feature importance algorithm developed in Section 3.4.1 is based on the Lipovetsky Shapley net effects. However, this algorithm is further extended to classification tasks by applying it to the logistic regression model. Model-agnostic approaches do not require any information about how the

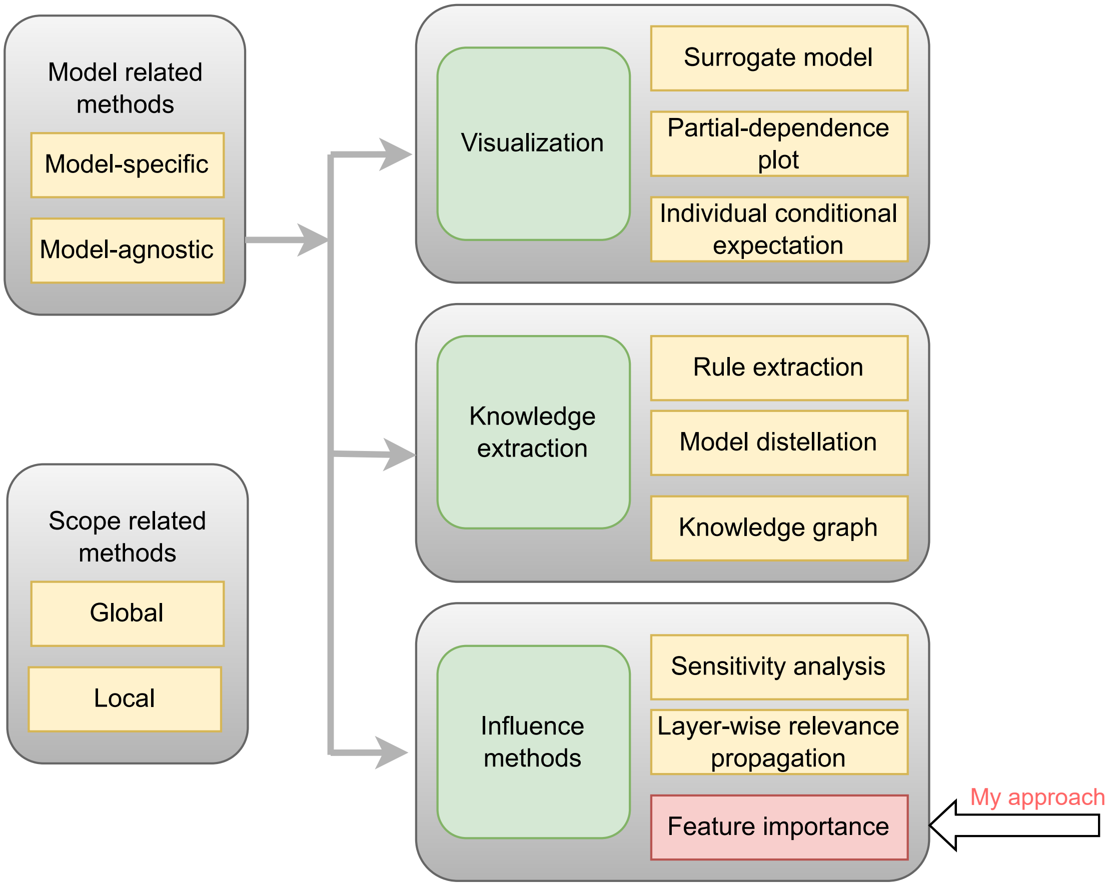


FIG. 2: Categories of XAI techniques

model makes predictions. Model-agnostic algorithms claim that feature selection algorithms could be applied to any model through the model's input and output. Model-agnostic approaches do not require any information about how the model makes predictions. The great benefit of model-agnostic interpretation methods over model-specific ones is model, explanation, and representation flexibilities. Other model-agnostic feature importance methods [72, 14] Shapley additive global importance (SAGE) and Relative Feature Importance (RFI) seem to overcome the limitations of model-dependent algorithms. However, these algorithms still need more tests and experiments to estimate the framework's robustness in the context of different datasets and researched problems.

Based on the scope, two types of explainability are categorized, i.e., global and local

[22]. Global scope refers to the explainability of the whole logic of the model, while local explainability tries to explain a specific instance or an individual prediction.

Based on the complexity models can be inherently explainable (regression model) or black-box (neural networks). Complexity and explainability are inversely related; complex the model is less explainable, the model outcomes are [73]. To have more explainable models, inherently explainable models, such as Bayesian rule lists [74], are developed.

The cooperative game theory-based XAI (CGTXAI) methods developed in the scope of this dissertation are model agnostic. Generally, these techniques are further classified into four categories: (i) Visualization, (ii) Knowledge extraction, (iii) Influence methods and (iv) Example-based explanation.

Visualization techniques help to reveal aspects that are difficult to observe from the black-box. Popular visualization techniques are surrogate models, Partial Dependence Plot (PDP), and Individual Conditional Expectation (ICE). A surrogate model is a simple explainable model used to explain a complex model [13]. The local interpretable model-agnostic explanations (LIME) [13] approach is a method for building local surrogate models to explain individual predictions. A partial dependence plot (PDP) shows the marginal effect one or two features have on the predicted outcome of a machine learning model [75]. Individual conditional expectation (ICE) is an extension of PDP, which displays one line per instance that shows the change of the instance prediction when the feature changes.

Knowledge extraction refers to generating understanding from structured or unstructured sources. Two main techniques for knowledge extraction are (a) rule extraction and (b) model distillation. Rule extraction is based on a symbolic description of information learned by the network during its training by obtaining rules that estimate the decision-making process [76]. Model distillation is a model compression to transfer information from deep networks to shallow networks [77].

Influence methods assess the importance of a feature by altering the input sequences and recording how much these changes affect model performance. Three major influence methods are (i) Sensitivity analysis, (ii) Layer-wise relevance propagation, and (iii) Feature importance. Sensitivity analysis refers to how model input or weight perturbations influence its output [78]. Sensitivity analysis is used to verify whether model output stays stable with purposefully perturbed data. Data perturbation and its respective visual demonstrations will strengthen the trust in the model outcomes. Layer-wise relevance propagation (LRP) is a technique that shows the explainability of highly complex deep neural networks by using purposely designed propagation rules and propagating the prediction starting from the output layer of the network and backpropagating up to the input layer [79]. Feature importance, the main focus of this dissertation, measures the contribution of each feature to the model performance when permuting the features. An example of a feature importance method is Shapley random forest measure [80], and Shapley additive global importance [14]. Feature importance is the key focus of this dissertation, and a more detailed description is presented in Subsection 2.2.1.

Each of these methods is better suited for certain problems and data. Sometimes, several methods are used to gain deeper insights into the problem. While these approaches are different and serve different purposes, a combination of these methods can be used to dive deeper into the data and gain better explainability. The next subsection presents feature importance methods, the main focus of this research.

2.2.1 FEATUE IMPORTANCE OVERVIEW

This section outlines the methods of explainable artificial intelligence.

Feature importance (FI), also called feature attribution [20] and credit valuation [21], is a crucial aspect of explainable artificial intelligence and refers to techniques that measure feature relevance values to better understand the data and the prediction [22]. Feature importance was first introduced by Breiman in 2001 [81]. Feature importance is calculated by removing the features and measuring the model's prediction performance. A feature is considered important if removing a feature, the model performance reduces. Feature importance models are described as model-specific and model-agnostic. Model-specific refers to the feature importance approach that is specifically designed for a particular modeling paradigm, such as random forest [81]. Model-agnostic algorithms claim the feature importance algorithms could be applied to any model through the model's input and output. Model-agnostic approaches do not require any information about how the model makes predictions. The great benefit of model-agnostic interpretation methods over model-specific ones is the model, explanation, and representation flexibilities [14, 72]. Feature importance techniques are further classified into two categories: local and global. Local techniques provide insights into the importance of individual features for specific instances or predictions, offering a detailed understanding of local model behavior. On the other hand, global techniques offer a broader perspective by quantifying the overall impact of features across the entire dataset, revealing their consistent influence on the model's performance.

Widely used feature importance methods that I considered in this study are SHAP (SHapley Additive exPlanations) [15], LIME (Local Interpretable Model-agnostic Explanations) [13], and permutation importance [82]. Below, each of these techniques is presented in detail.

SHAP (SHapley Additive exPlanations)

Shapley additive explanations (SHAP) [15] is a model-agnostic approach based on cooperative game theory that delivers insights into the rationale behind a model's predictions. Lundberg and Lee [15] identify a new class of additive feature importance measures and suggest new methods that show improved computational performance and/or better consistency with human intuition than previous approaches. This methodology assigns importance values to each feature, indicating their contribution to the prediction process. SHAP values capture the average marginal contribution of a feature across all possible feature combinations. By considering all possible feature subsets, Shapley additive explanations provides a comprehensive understanding of feature importance and interactions. The foundational core of Shapley additive explanations lies within the Shapley value equation [83], presented as follows: $\phi_i(f) = \sum_{S\subseteq N\setminus\{i\}} \frac{|S|!(|N|-|S|-1)!}{|N|!}[f(S\cup\{i\}) - f(S)]$. Where, $\phi_i(f)$ represents the Shapley value for feature i and model f. N is the set of all features. S is a subset of features excluding i. $|S|$ is the number of features in subset S. $f(S\cup\{i\})$ is the model prediction when including feature i along with subset S. $f(S)$ is the model prediction when considering only subset S without feature i. More information on Shapley values and cooperative game theory is presented in Section 2.3.

While SHAP stands as a pioneering technique, researchers have endeavored to enhance certain facets of its functionality. One such advancement is KernelSHAP, which presents an alternative approach inspired by local surrogate models [84]. KernelSHAP offers an innovative kernel-based estimation of Shapley values, enhancing the precision of feature importance attribution. In parallel, TreeSHAP was developed as an efficient estimation strategy tailored specifically for tree-based models, further broadening the horizons of SHAP's applicability and usability [15].

Shapley additive explanations could be used in numerous ways to gain insights into various facets of predictions, including explanations of individual instance effects, summaries of overall feature contributions, analyses of dependencies, and detailed force plots of feature impacts.

SHAP: Local explainability

In the following, Shapley additive explanations (SHAP) plots (waterfall, force plot) are presented to provide localized explanations for feature importance values. SHAP plots can be generated using the Python shap library. Gaining a thorough grasp of these visual representations is crucial for effectively interpreting model behaviors.

SHAP waterfall plot offers a visual representation of how individual feature values contribute to a model's prediction for a specific instance. It generates a baseline value, which is at the top of the plot. This represents the expected output of the model before considering any feature contributions. Generally, this baseline aligns with the model's overall average prediction. Every subsequent bar within the plot symbolizes the precise contribution of a distinct feature to the model's prediction. These features are arranged in descending order based on their absolute Shapley values for that particular instance. Positive contributions, depicted in red, indicate that the presence of a feature elevates the prediction. Conversely, negative contributions, depicted in blue, signify a feature's diminishing effect on the prediction. The length of each consecutive bar corresponds to the extent of a feature's impact on the final prediction. The cumulative effect of all feature contributions equates to the disparity between the baseline value and the ultimate model prediction. At the plot's lowermost point, we encounter the model's final prediction, a culmination of the baseline value and the combined influence of individual feature contributions.

SHAP force plot are equivalent representations of waterfall plots that display the key information in a more condensed format. This plot as well presents the magnitude and direction of a feature's influence on a particular prediction. Force plots are useful in simultaneously presenting explanations across numerous instances of the dataset, facilitating straightforward comparisons.

SHAP: Global explainability

In this section, a range of Shapley additive explanations (SHAP) global explainability plots are presented. This includes a bar plot illustrating the mean absolute SHAP values, beeswarm plots, a SHAP heatmap, SHAP dependence plots, and SHAP partial dependence plots.

SHAP bar plot of Mean absolute SHAP values are commonly used to visualize bar plots that arrange features based on their significance. The mean absolute SHAP values are, on average, how much each variable impacts the prediction. This estimate helps to understand both the sequence of features and the proportional intensities of the mean absolute SHAP values.

SHAP beeswarm plots present a sophisticated and information-rich representation of SHAP values, unveiling not only the relative importance of features but also their actual interactions with the predicted outcomes. In a beeswarm plot, each feature corresponds to a unique point for every instance in the dataset, distributed horizontally along the X-axis based on their SHAP values. When high SHAP value densities occur, the points vertically stack, conveying the concentration. The color bar represents the variable's row values for each instance on the graph, distinct from the SHAP values themselves. Elevated feature values for a particular instance manifest as red dots, while lower values are depicted as blue dots. By examining the color distribution horizontally along the x-axis for each feature, insights into the intrinsic relationship between data and their corresponding SHAP values could be obtained.

SHAP heatmap provides a visual summary of the impact of different features on the predictions made by a machine learning model. Each row in the heatmap corresponds to an instance, and each column represents a specific feature. The cells of the heatmap are colored based on the corresponding SHAP values, often using a color scale to indicate the magnitude and direction of the impact. Positioned above the heatmap matrix is the model's output, while on the right-hand side, a bar plot illustrates the overall significance of each model input in a global context.

SHAP dependence plots help to better understand the relationship between a feature's values and the model's predicted outcomes. In the SHAP dependence plot, instances are presented as a scatter plot. The horizontal axis represents the feature's value, while the vertical axis displays the SHAP value assigned to that feature. This SHAP value indicates the extent to which the model's prediction for a specific sample is influenced by the feature's value. The color variation is linked to a different feature, potentially showcasing an interaction effect with the main feature being plotted. The SHAP dependence algorithm inherently selects this secondary feature. If an interaction effect exists between the primary feature and the second feature, an observable color pattern emerges.

SHAP partial dependence plots (PDP) illustrate the marginal influence that one or two features exert on the predicted outcome of a machine learning model, as highlighted by Friedman [75]. The partial dependence plot explains the connection between the selected feature and the target variable, enabling the identification of linearity, monotonicity, or any associations [57].

In summation, SHAP is a dynamic methodological framework, unraveling the intricate nature of machine learning models through its cooperative game-theoretic foundations. It bestows interpretability and transparency, empowering practitioners to decipher the nuanced contributions of features and interactions within their models, thereby fostering a deeper understanding of complex predictions.

Alternative to SHAP is SAGE (Shapley additive global importance) which is an approximation algorithm that handles the NP problem of subset generation with Shapley value [14]. To evaluate the cooperative game with sampling, the method suggests sampling the removed features from their joint marginal (rather than conditional distribution) and then averaging the model output. The method suggests random sampling permutations of the features and measuring each feature's marginal contribution to that ordering to determine the Shapley value. Similar to SHAP, SAGE is a model-agnostic approach and could be applied to any dataset and problem.

LIME

LIME (Local Interpretable Model-Agnostic Explanations) [13] is a model-agnostic method that explains the predictions of black-box models by approximating its behavior locally. It creates surrogate interpretable models around individual instances to estimate their feature importance. LIME generates locally weighted explanations by perturbing the features and measuring the impact on the model's predictions. This enables an understanding of the model's decision-making process at a local level.

$$\text{Explanation}(x) = \arg\min_{g \in G} L(f, g, \pi_x) + \Omega(g) \tag{1}$$

Where f is the black-box model being explained, g is the surrogate model, $L(f, g, \pi_x)$ is a loss function that quantifies the difference between $f(x)$ and $g(x)$ for perturbed instances sampled from a distribution π_x. $\Omega(g)$ is a regularization term that encourages the surrogate model g to be simple. G is the set of possible surrogate models. By constructing interpretable surrogate models proximate to specific instances, LIME offers insights into feature importance. These surrogate models, often simpler linear regressions or decision trees, approximate the complex black-box model behavior, enhancing human comprehensibility. Leveraging perturbation and sampling, LIME measures feature impact by observing prediction variations when features are perturbed.

Permutation importance

Permutation importance (PIMP) [82] is a feature importance technique that measures the impact of shuffling a feature's values on the model's performance. Permutation importance employs repeated shuffling of the outcome vector to estimate the importance distribution for each feature in the dataset. Then the P-values from this distribution are computed, which offers an adjusted evaluation of feature significance. Permutation importance shows that features lacking informative values do not obtain significant P-values. Therefore this method helps to distinguish informative from non-informative features. Machine learning models are constructed using features identified as statistically significant, and their performance is evaluated by comparing models that utilize statistically significant features against those incorporating original features. Here, original features refer to all features prior to undergoing permutations and the assessment of their importance, values, and significance.

Permutation importance evaluates the decrease in model performance (e.g., accuracy or error) when a feature's values are randomly permuted. The larger the decrease, the more important the feature is considered. Permutation importance provides a simple yet effective way to assess the relative importance of features in a model. Mathematically, permutation importance can be expressed as follows:

Permutation importance quantifies the change in model performance due to the permutation of a feature, thereby revealing the feature's contribution to the model's predictive power.

2.3 GAME THEORY

This section briefly describes what game theory is, focusing on cooperative game theory and its solution concepts. This section was developed based on a Grigoryan and Collins [85] paper written during this dissertation.

Game theory is the process of modeling strategic interactions between two or more players in a situation where competition and conflict prevail. Games are based on a set of rules, and the players' preferences over their possible strategies lead to the outcomes of the game. Game theory was first introduced by Antoine Cournot in 1838, but the modern study came after 1944 when Von Neumann's and Morgenstern published their famous book: the theory of games and economic behaviour [86]. Classic examples of where game theory can be used in real-life situations include chess, wargaming, auction design, and business negotiations [87, 88]. Games are made up of players who wish to choose a strategy to maximize their payoff. The players are strategic decision-makers within the context of the game. Players can be human and non-human (e.g., organization, vehicle, mobile nodes) members of the game. A strategy is a complete plan of action a player may take, given the set of circumstances within the game. Even though game theory is primarily used in mathematics and economics [89], it has made a significant impact on a large number of disciplines ranging from politics, science, biology, psychology, sociology, computer science, and engineering [90]. Robert Aumann and Oliver Hart explain the interdisciplinary use of game theory in the following way: "Game Theory may be viewed as a sort of umbrella or 'unified field' theory for the rational side of social science, where 'social' is interpreted broadly, to include human as well as non-human players (computers, animals, plants) ... It does not use different ad hoc constructs ... It develops methodologies that apply in principle to all interactive situations" [91]. This statement illustrates the interdisciplinary and broad nature of game theory to model, predict and explain different phenomena and situations of interest that involve multiple decision-makers.

According to the constraints and situations of the games, game-theoretic models can be grouped into several categories. To study conflict resolution and strategic cooperation from a Systems Engineering perspective, Hipel and Obeidi [92] suggest a classification of the game theory as a non-quantitative and a quantitative approach.

Non-quantitative approaches assume relative preference information, such as one action being more preferred or equally preferred to another. In this case, a player does not have to know precisely how much one action is preferred over another. Hipel and Obeidi [92] brought the following example to describe a non-quantitative approach: a marketing agent has to suggest whether a car having a sleeker aerodynamic design would be preferable to young people than another model with functional "square" design. The agent may reply that a more elegant design would be preferred but will not be able to specify cardinal numbers to represent preferences. Based on Figure 1, metagame analysis is a subdivision of the non-quantitative approach [93], which later was expanded into conflict analysis. This expansion had a significant contribution to the development of the Graph Model for Conflict Resolution. Drama theory, developed by Bryant [94], is considered another non-quantitative technique. It describes situations in terms of subjective frames: games that can change as a result of the internal pressure created by the interaction of characters, i.e., players.

Hipel and Obeidi [92] state that quantitative methods assume cardinal preference information, such as cardinal utility values. By this, they suggest that decision-makers use real numbers to model preferences. Quantitative games are, by far, the most common game theory approach; many popular textbooks on game theory do not even mention non-quantitative approaches [95]. The quantitative approach is further classified into non-cooperative, i.e., normal-form, extensive-form, etc., and cooperative games. Non-cooperative game-theoretic method refers to games with competition between individual players and tries to predict the player's individual strategies and payoffs.

The second classification of games is cooperative (coalitional) games, presented in more detail in section 2.3.1. They study the behavior of a rational player when they cooperate to form coalitions [90]. As such, cooperative games consider three or more players. This approach mainly focuses on predicting which alliances will form. Cooperative games can be subdivided into games with transferable utility games and games with non-transferable utility. Despite the differentiation between nonquantitative and quantitative approaches, Hipel and Obeidi [92] emphasize that both these approaches constitute mathematical models.

A non-cooperative game specifies all the possible actions for each player, and their main goal is to maximize their payoffs [96]; solution mechanism focus on strategies of each player that satisfy some solution criteria. A well-known solution concept in non-cooperative game theory is the Nash Equilibrium. Other examples of solution concepts are sub-game perfect equilibrium and correlated equilibrium introduced by Bielefeld [97] and Aumann [98], respectively. In non-cooperative game theory, strategies can be defined as either pure or mixed. A pure strategy is when a deterministic action is chosen, whereas, a mixed strategy is when the action is randomly selected using a probability distribution over the pure strategies. The payoff is the payout a player receives after reaching a particular outcome. The outcome of the game-theoretical model is that all players have made their decisions. A solution is an outcome that satisfies some criteria. A basic solution concept in game theory is the Nash Equilibrium in which where each player maximizes his payoff with respect to his own strategy choice, given the current strategy choices of other players [99].

2.3.1 COOPERATIVE GAME THEORY

This section delves into the principles of cooperative game theory.

A cooperative (coalitional) game theory helps to study the behavior of rational players when they cooperate to form coalitions [90]. This approach mainly focuses on predicting which alliances will form. This is the main approach to situations with three or more players. The cooperation can be explained by the rational choice of self-interested players rather than by altruism [83, 85, 100, 101]. In cooperative games, the players can form coalitions to achieve a better payoff. Bargaining games and coalition formation games are considered subcategories of cooperative games [102]. Note that bargaining games have been discussed from both non-cooperative and cooperative game theory perspectives [103]. In a non-cooperative bargaining game, players individually try to maximize their utility without regard to the utility achieved by other players. In contrast, in a cooperative bargaining game, players bargain and coordinate their actions before the game is played. Consequently, players act according to the agreement reached. The agreements reached must be binding, so players are not permitted to deviate from their agreement. Players act in a non-cooperative way if they cannot reach an agreement. Saad et al. [104] classify coalitional games into three categories: canonical coalitional games, coalition formation games, and coalitional graph games. There are subtle differences between the different forms, which we will not discuss in this paper.

An essential concept in cooperative game theory is a coalition, which refers to the formation of sub-sets of players. The value, v, of a coalition is the total payoff that the coalition members can guarantee themselves collectively. In some cooperative games, players can transfer utilities they get to other members of their coalition (transferable utility games). In others, this transfer is impossible (non-transferable utility games) [105].

Cooperative games can be subdivided into games with transferable utility (TU) games and games with non-transferable utility (NTU). A particular case of coalitional games is conflicting claims problems [106].

Cooperative game theory answers the following two questions:

1. Which coalitions will form?

2. How to divide the coalitions' payoff among the players?

Answering two questions at once can be problematic, so many games developed put some restrictions to reduce this down to one question.

An example of cooperative game theory in academia is the collaboration between a Ph.D. student and an advisor. Both parties aim to form a coalition that maximizes their respective outcomes. For the student, the ideal outcome might be gaining knowledge, skills, and a path to successful completion of their dissertation. For the advisor, the outcome might be advancing their research agenda, enhancing their reputation, or contributing to the academic community. Once the coalition is formed, the coalition members, i.e., the advisor and the student, allocate efforts (research tasks, teaching, academic writing) and rewards (authorship on papers, presentations at conferences, academic and professional recognition). The value (or utility) generated by their collaboration is measured in terms of "impact points" that represent academic recognition, future funding potential, and contribution to the academic community. Assume the following:

- Alone, the student can generate 10 impact points through their independent research efforts.
- Alone, the advisor, given their experience and network, can generate 20 impact points by guiding other projects or their own research.
- Together, through combined efforts, they can generate 40 impact points due to the advisor's guidance and the student's work.

By considering cooperative game theory, the objective is to determine how to fairly divide the 40 impact points between the student and advisor using a cooperative game theory called Shapley value (described in detail in Section 2.3.2). Shapley values [24] suggests that members should receive payments or shares proportional to their marginal contributions. The steps to compute the Shapley value-based distribution of the 40 impact points for this example are as follows:

1. List all permutations of the coalition (in this case, there are only two members, so there are two permutations):
   - Student first, then advisor.
   - Advisor first, then student.
2. Calculate the marginal contributions:
   - When the student comes first, their contribution is 10 (since they can generate 10 points alone). The advisor's marginal contribution is 30 (a total of 40 minus the 10 points the student could achieve alone).
   - When the advisor comes first, their contribution is 20. The student's marginal contribution is 20 (the total of 40 minus the 20 points the advisor could achieve alone).
3. Compute the Shapley value for each coalition member
   - Student's Shapley value = [(Marginal contribution when a student comes first) + (Marginal contribution when coming after the advisor)] / 2 = $\frac{(10+20)}{2} = 15$ impact points
   - Advisor's Shapley value = [(Marginal contribution when advisor comes first) + (Marginal contribution when coming after the student)] / 2 = $\frac{(20+30)}{2} = 25$ impact points

According to the Shapley value, the fair division of the 40 impact points for the combined research efforts would allocate 15 impact points to the student and 25 impact points to the advisor. This reflects a fair allocation based on each party's contribution to the collective outcome.

In a cooperative game theory, if superadditivity is assumed, that is $S, T \subset N,$ if $S \cap T =$ then $v(S \cup T) \geq v(S) + v(T)$, then it can be shown that the grand coalition will form (i.e., the coalition containing all players) and, hence, the only concern is how to split the payoff among the players. A non-transferable utility game called a hedonic game assumes that each player will receive a single fixed payoff from any coalition they are a member [107, 108]. Hence, hedonic games only need to consider the second question. Hedonic games are a specific class of cooperative games [109]. These games are used to model situations where the primary concern is the individual preferences or happiness of players when forming coalitions. Collins et al. [100] have developed a repeated generation of a random hedonic game and determined the core set. The experiment was repeated for a different number of players in a game, ranging from three to seven. The results from games of one or two players can easily be solved analytically and have been included in the results for completeness. It was found that having a single core (solution) was the most common result for a game.

Shapley [24] stated that members should receive payments or shares proportional to their marginal contributions. Stable outcomes, or the core of a game, are the outcomes that no new coalition could form where all its members do better than their current coalitions [110]. However, the core is not the only solution mechanism of a cooperative game. Over the years, researchers have proposed different solution concepts, such as the Shapley value, the core, the kernel, and the nucleolus [111]; they are all based on different notions of fairness and stability.

In the machine learning context, the cooperative game embodies the ML model, and the game participants represent the feature space. Subsets of participants are called coalitions, and the setting where the utilities (payoffs) are given to these coalition members is known as transferable utility games (TU). Games where the transfer of utilities is impossible are non-transferable utility (NTU). Coalitional games are further categorized into canonical coalitional games, coalition formation games, and coalitional graph games [104]. In canonical games, no group of players can do worse by joining a coalition than acting non-cooperatively. In coalition formation games, forming a coalition brings advantages to its members, but the cost of forming the coalition limits the gains. Coalitional graph games are presented in a graph form, and the interconnection between the players significantly affects the outcome of the game.

Notation

A cooperative game in characteristic function form is defined as a $2-$tuple $(N, v), N$ being the set of players $N = \{1, 2, 3, ..., n\}$. The players form coalitions, which refers to the formation of sub-sets of players $C, S, X \subseteq N$. For a set $A : C_A$ denotes the subsets of A, i.e., $C \subseteq A$, and P_A denotes the partitions of A. For a set of players N, a coalition is any subset of N, and N is the grand coalition, which contains all players. A partition of N is the splitting of all the players into disjoint coalitions. The value $v : C_N \to \mathbb{R}$ of a coalition is the characteristic function, and for each coalition of players $C \subseteq N, v(C)$ is the total payoff that the coalition members can guarantee themselves, collectively, and it satisfies the $v(\emptyset) = 0$. Also v is assumed non-negative $v(C) \geq 0$, for any $C \subseteq N$, and monotone: $v(C) \leq v(D)$, for any C, D such that $C \subseteq D$. An outcome of a game $\Gamma = (N, v)$ is a pair (P, x), where: $P = (C_1, C_2, ..., C_k) \in P_N$ is a coalition structure (CS), and $x = (x_1, ..., x_n)$ is a payoff vector, which distributes the value of each coalition in P. Another important concept in CGT is imputation. An efficient payoff vector is called pre-imputation, and an individual rational pre-imputation is called imputation. Imputation is a vector that assigns how much payoff goes to each of the players [112]. Imputation is defined as a vector of $a = (a_1, ..., a_n)$ satisfying the following conditions:

1. Individual rationality, which indicates that player i should receive no less than it receives alone: $a_1 \geq v(\{i\}), i \in N$
2. Group rationality indicates that the whole payoff that the grand coalition earns should be allocated among the players: $\sum_{i=1}^{n} a_i = v(N)$.

A subclass of games in the characteristic form consists of superadditive games, that is $S, T \subset N$, if $S \cap T = \emptyset$ then $v(S \cup T) \geq v(S) + v(T)$, then it can be shown that the grand coalition will form (i.e., the coalition containing all players). An important subclass of superadditive games is the convex game. The convexity is a stronger condition than superadditivity, and the game is convex if for all $S, T \subset N, \nu(S \cup T) \geq \nu(S) + \nu(T) - \nu(S \cap T)$

Technically, a cooperative game can be represented as a non-cooperative game in certain circumstances [113]. With this representation, the Nash Equilibrium could be found as a solution concept. However, in practice, this approach becomes computationally intractable.

Next, the cooperative game theory solution concepts and some of their properties are described. The methodology section (Section 3) illustrates the use of these solutions to measure feature importance methods considering linear and logistic regression models.

Transferable utility (TU) games, Shapley values, core, and Nucleolus are presented first (Subsection 2.3.2), followed by voting games (Subsection 2.3.3), and finally, conflicting claims solutions are presented in Subsection 2.3.4.

2.3.2 TRANSFERABLE UTILITY GAMES

Transferable Utility (TU) game theory is a branch of cooperative game theory where it is assumed that utility (payoff value) can be transferred between players without loss [114]. Under TU, the primary focus becomes identifying the total value a coalition can generate and determining ways to distribute this value among its members. These ways refer to the established solution concepts, such as Shapley values, core, or Nucleolus, that divide the total value or utility generated by a coalition among its members in a cooperative game setting under the assumption of transferable utility [115, 90].

For an example of transferable utility cooperative game theory, imagine a software development company that has decided to undertake a new project to develop a new product. The product development requires a combination of skills: coding, design, and sales. The company needs to form a team that includes an Engineer (E), a Designer (D), and a Sales Representative (S). Each of these professionals can contribute to the project's performance, but their full potential is realized when they work together due to the complementary nature of their skills. Engineer (E) can contribute $40,000 worth of value to the project. Designer (D) can contribute $30,000 worth of value. Sales representative (S) can contribute $20,000 worth of value. Working independently, they contribute their respective values, but when they collaborate, they enhance the project's total value due to their complementary efforts. When working together E and D together generate $100,000, E and S together generate $80,000, D and S together generate $70,000. E, D, and S together (grand coalition, i.e., the full team) generate $150,000. The total value generated by the full team (E, D, S) working together is greater than the sum of their individual contributions due to the synergistic effect of their collaboration. This reflects the group rationality discussed above. With transferable utility cooperative game theory, the goal is to determine how to divide (allocate) the $150,000 in a way that fairly compensates each coalition member for their contribution, including the added value from their cooperation. The ability to allocate the $150,000 among the coalition members, despite their individual capabilities to earn individually, demonstrates transferable utility.

The counterpart to a transferable utility game is a non-transferable utility game, where the utility or benefits generated by the coalition cannot be easily divided or allocated among its members. For instance, when the utility derived from cooperation is intangible or indivisible, such as the reputation gained by coalition members, dividing this benefit fairly poses a significant challenge. Unlike monetary rewards that can be distributed in precise proportions, reputation enhancement lacks a straightforward method for division. This discrepancy highlights the complexity inherent in non-transferable utility scenarios, where the benefits of collaboration extend beyond quantifiable gains, requiring nuanced approaches to ensure fair recognition and reward within the coalition. Non-transferable utility games are not considered in the scope of this dissertation.

The following section presents three key solution concepts within the framework of transferable utility that could be used to solve transferable utility games: the Shapley Value, the Core, and the Nucleolus.

Shapley value

First, the solution of the Shapley value is presented.

Shapley suggested that members should receive payments or shares proportional to their marginal contributions by considering arbitrary permutations of the set N [116].

$$\phi_i(v) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} (v(S \cup \{i\}) - v(S)) \tag{2}$$

Shapley value states we shall consider an arbitrary permutation P of the ordered set of the players. The formula permits the presentation of Shapley axiomatics. First, we set up a correspondence between every cooperative game N, v and the vector $\phi_{(i)} = (\phi_{(1)}[i], ..., \phi_{(n)}[i])$, whose components are interpreted to mean the payoffs received by players under an agreement. $|N|$ is the total number of players. Here, this correspondence is taken to satisfy the following axioms [117].

1. Efficiency: The sum of the cost or values of all agents equals the value of the grand coalition so that all the gain is distributed among the agents, i.e., $\sum_{i \in N} \phi(i) = v(N)$. This assumes superadditivity within the game and indicates that what each player receives must be equal to what the grand coalition has produced.

2. Symmetry: If i and j contribute the same to all coalition subsets S, they should receive the same share, represented as $v(S \cup \{i\}) = v(S \cup \{j\}) \quad \forall S \subset N; \quad i, j \notin S$, such that $\phi(i) = \phi(j)$, and i, j are symmetric with respect to each other.

   This axiom implies that if features X_1 and X_2 consistently make equal contributions to all sub-coalitions, then both features should receive an equal share of the performance or the output from the model.

3. Dummy player: If the cost of the i-th player does not contribute to the total cost of the coalition in a cooperative game, i.e., if $v(S \cup \{i\}) - v(S) = 0, \quad \forall S \subset N$, then such

players are called a dummy or null players, and $\phi(i) = 0$. This implies that the absence of contribution leads to receiving nothing.

4. Linearity: If v_k and v'_k are two characteristics functions of a coalition game, then $\phi_{v+v'}(i) = \phi_v(i) + \phi'_v(i)$, where $\forall S(v + v')(S) = v(S) + v'(S)$. This axiom states that the values from the games can be combined in an additive way.

   These axioms suffice to define unique values for coalitional games.

Example: Suppose that there are only three players in the dataset and the characteristic functions for the three players are as follows: $v(\{0\}) = 0, v(\{1\}) = 0.15, v(\{2\}) = 0.3, v(\{3\}) = 0.04, v(\{12\}) = 0.5, v(\{13\}) = 0.6, v(\{23\}) = 0.62, v(\{123\}) = 0.88$. For a three-player scenario, there will be 8 permutations.

TABLE 1: Shapley Value computation example with 3 features

Permutations	**Variable 1**	**Variable 2**	**Variable 3**
123	$v(\{1\}) = 0.15$	$v(\{1,2\}) - v(\{1\}) = 0.35$	$v(\{1,2,3\}) - v(\{1,2\}) = 0.38$
132	$v(\{1\}) = 0.15$	$v(\{1,2,3\}) - v(\{1,3\}) = 0.28$	$v(\{1,3\}) - v(\{1\}) = 0.45$
213	$v(\{1,2\}) - v(\{2\}) = 0.2$	$v(\{2\}) = 0.3$	$v(\{1,2,3\}) - v(\{1,2\}) = 0.38$
231	$v(\{1,2,3\}) - v(\{2,3\}) = 0.26$	$v(\{2\}) = 0.3$	$v(\{2,3\}) - v(\{2\}) = 0.32$
312	$v(\{1,3\}) - v(\{3\}) = 0.56$	$v(\{1,2,3\}) - v(\{1,3\}) = 0.28$	$v(\{3\}) = 0.04$
321	$v(\{1,2,3\}) - v(\{2,3\}) = 0.26$	$v(\{2,3\}) - v(\{3\}) = 0.58$	$v(\{3\}) = 0.04$
Shapley Values	0.263	0.348	0.268333

The Shapley value calculation suggests the following allocation for players 1, 2, and 3 $\phi(i) = (0.263, 0.3483, 0.2683)$. These values indicate that player 2, an equivalent of 0.348, has the highest marginal contribution. player 1 and player 2 respectively equal 0.263 and 0.268, which suggests that players 1 and 3 share almost equal contributions.

Core

The next solution concept presented is the core. This is tightly related to the imputation concept introduced in Section 2. Specifically, imputation dominance is essential when determining the core of the game. Imputation α dominates the imputation β by coalition S (notion $\alpha \succ_S \beta$), if $\alpha > \beta_i, i \in S, \sum_{i \in S} \alpha_i \leq v(S)$. Let's consider two example imputations:

$\alpha = (40, 35, 40), \beta = (50, 30, 35)$. From this imputation, there exists coalition $\{2, 3\}$, where α is dominated, i.e., $\alpha \succ_{2,3} \beta$. This could be verified with the following computation:

$$\alpha_2 = 35 > 30 = \beta_2$$

$$\alpha_3 = 40 > 35 = \beta_3$$

$$\alpha_2 + \alpha_3 = 75 \leq 75 = \nu(S)$$

The set of all nondominated imputations of a cooperative game is called the Core [110]. Imputation α belongs to the core, if and only if: (N, ν)

$$\sum_{i \in S} \alpha_i \geq \nu(S), S \subset N, and \sum_{i \in S} \alpha_i = \nu(N) \tag{3}$$

A stable outcome, or the core of a game, is the outcome that no new coalition could form where all its members do better than their current coalitions [110]. In other words, the core refers to the set of efficient payoff vectors such that no coalition can achieve a better payoff by itself.

Mathematically, the core is computed as a set that satisfies a system of weak linear inequalities. The core is closed and convex, characterized by increasing marginal utility for coalition members as coalitions grow larger. It is possible a situation where the core of the game is empty, meaning no stable coalition exists. For example, if one unit of a good should be shared among a coalition having at least $\frac{(n+1)}{2}$ members, where n is an odd number that has an empty core. The Bondareva–Shapley theorem states the core of ν is nonempty if and only if ν is balanced [118, 119]. For the sake of brevity, the balanced condition of the game is not presented; instead, the original paper is referred to for its detailed description [118, 119].

Nucleolus

The nucleolus is another efficient method to determine a fair division of the payoff among coalition members [111, 112]. The nucleolus is the set of efficient and individually rational vectors, that is, the gain that players in coalition S can obtain if they leave the grand coalition N under the imputation x and instead take the payoff $v(S)$. The nucleolus satisfies the first three axioms of the Shapley value and has some advantages over it. The objective function of the nucleolus solution is to make the coalitions' excess (the largest unhappiness) as small as possible or, equivalently, minimize the worst inequity. Schmeidler [120] defines "unhappiness" or excess of a coalition as the difference between what the members of the coalition could get by themselves and what they are actually getting if they accept the allocations suggested by a solution. Excess, an inequity measure of an imputation (allocated payoff) x for a coalition S, is defined as:

$$e(x, S) = v(S) - \sum_{j \in S} x_j \tag{4}$$

Excess $e(x, S)$ measures the amount or the size of the inequity by which coalition S falls short of its potential $v(S)$ in the allocation x. During the distribution of worth, the coalition that "complains" that it is not getting its proper share efforts will be made to give it a fair share.

The steps of finding the nucleolus are to find a vector $x = (x_1, x_2, ..., x_n)$ that minimizes the maximum of Z the excesses $e(x, S)$ over all S subject to $x_i = v(N)$ The process of minimizing the maximum (min-max) of a collection of linear functions subject to a linear constraint is converted to a linear programming problem. Second and more linear programming problems may be used to minimize the next largest excess until n-tuple imputation x is found.

$$\text{min } Z \text{subject to:}$$

$$Z + \sum_{i \in S} x_i \geq \nu(S) \quad \forall S \subseteq N \tag{5}$$

$$\sum_{i \in S} x_i = \nu(N)$$

The linear programming problem is optimized based on the Equation 5.

2.3.3 VOTING GAME

This section explores the core concepts of voting games, with a particular focus on the Shapley-Shubik and Banzhaf power indices.

Voting games represent another configuration within the realm of coalition games, often featuring the presence of a pivotal or veto player. These games are mathematical models for exploring scenarios in which participants collaborate to form coalitions with the objective of reaching or surpassing a specific threshold, often called a quota (denoted as q)[121]. The specific value of the quota can vary depending on the voting system in use. The success or the winning of the coalition is determined based on the preferences and influence of a subset of players whose combined weights meet or exceed the quota. This is obtained and measured by their respective weights [121]. A well-designed voting system should be fair and transparent, described with a player set that includes all the parties participating in the voting game [121].

Voting games $(N, w_{i \in N}, q)$, noted as $[q; w_1, ..., w_n]$ takes the following form:

$$\nu(S) = \begin{cases} 1, & \text{when } \sum_{i \in S} w_i \geq q \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

Here, w_i is the number of votes of player $i, i \in N$, in other words, it shows the weight of the player i in the system. q is the threshold of votes. The decision is considered affirmative if the number of votes for this particular decision is more than the threshold. Equation 6 describes two cases. First, the characteristic function of coalition S is 1, or coalition S is winning if the sum of weights in coalition S is more or equal to the value of the threshold. The characteristic function of coalition S is zero or losing, so it cannot make the affirmative decision in the system if the number of votes from coalition S is strictly less than the threshold q. When $v(S) = 1$ and $v(S - i) = 0$, the player i is considered critical with respect to S in characteristic function v, and S is a pivot for i in $v()$. $P_i(v)$ can be used to denote the collection of pivots for i in $v()$.

In the next subsections, I will present two weighted voting games: the Shapley-Shubik power index and the Banzhaf power index.

Shapley-Shubik index

Shapley-Shubik analyzes situations where the power of the coalition might not be proportional to the size of the coalition, but it could be measured by the fraction of the possible voting sequences [122]. This refers to situations where the order in which participants join the coalition is crucial, and we are interested in determining the power of the participants in the system.

The Shapley-Shubik power index can be expressed as:

$$\phi_i = \sum_{S \in P_i(v)} \frac{(s-1)!(n-s)!}{n!} \quad \left(with \sum_{i \in N} \phi_i = 1 \right) \tag{7}$$

where $s = |S|$ is the number of voters in set S. The summation is taken over all winning coalitions S for which S without i, $S - \{i\}$ is losing. The Shapley-Shubik determines the number of sequences in which player i is pivotal over all possible orderings of n players. A player i is pivotal or swing for a coalition S if the player i turns S from losing $v(S) = 0$ to a winning coalition $v(S) = 1$ by joining that coalition.

The Shapley-Shubik power index is based on the Shapley value and satisfies many of its properties, such as efficiency, linearity, dummy, and symmetry. The difference is that it is defined using the characteristic function described in Equation 6 for the voting games so that it can be only zero or one. The sum here is taken by the coalition S, which do not include player i; without the player i are losing, and with player i are winning coalitions.

In a nutshell, the Shapley-Shubik power index can be conceptualized as the probability that a player is pivotal, given the assumption that all permutations are equally likely to occur.

Banzhaf Power index

Banzhaf power index (BPI) focuses on evaluating the power or importance of each player in a weighted voting system. BPI was first discussed by Lionel Penrose in 1946 [123] but was reintroduced by John Banzhaf in 1965 [124]. The BPI would be a numerical representation of how likely the player is to be critical, substantially influence the final decision, and control the outcome.

Banzhaf power index $\beta_i(\Gamma)$ originally was used to assess the power of players in a simple game [124]. The simple game $\Gamma = (N, \gamma)$ is a cooperative game such that $v(S) = 1$ or 0 for all S, and $v(N) = 1$ due to satisfying the superadditivity. BPI is similar to the Shapley-Shubik index and shares the same characteristics as weighted voting games. However, unlike the Shapley-Shubik index, BPI assumes that all player combinations are equally likely.

Banzhaf power index $\beta_i(\Gamma)$ is specified as

$$\beta_i(\Gamma) = \frac{1}{2^{n-1}} \sum_{S \subseteq N \backslash \{i\}} \Delta_i(S), \tag{8}$$

where, 2^{n-1} refers to the total number of player subsets $S \subseteq N\backslash\{i\}$ and $\Delta_i(S)$ is the marginal contribution of player i, $\Delta_i(S) = v(S \cup \{i\}) - v(S)$

2.3.4 CONFLICTING CLAIMS PROBLEM

In this section, solutions to conflicting claims are presented.

Game theory bargaining solutions to conflicting claims problems (also known as a bankruptcy problem) is a particular case of the distribution problem, in which the amount to be distributed, the endowment E is not enough to satisfy the players' claims on it [125]. A classic scenario involves allocating funds from a bankrupt company to its creditors, where conflicting claims solutions tackle dilemmas such as determining the equitable distribution of a bankrupt firm's liquidation assets among its creditors [126, 127]. The applications of conflicting claims theory extend across various domains. This includes allocating medical resources, determining budget allocations in educational institutions [128], or distribution of global emission budget [129] and allocation of fishing quotas[130].

A vast number of solutions [115] have been developed for solving conflicting claims problems, being proportional, constrained equal awards (CEA), constrained equal losses (CEL), Talmud (T), and random arrival (RA).

Conflicting claims problem consists of players $N = 1, 2, ..., n$ and amount $E \in \mathbb{R}_+$ of an infinite divisible resource, the endowment, that has to be allocated among them. Each player has a claim, $c_i \in \mathbb{R}_+$ on it. The $c = (c_i)_{i \in N}$ is the claims vector.

A conflicting claims problem is (E, c) with $\sum_{i=1}^{n} c_i > E$. The players are ordered according to their claims, $c_1 \leq c_2 \leq \cdots \leq c_n$, and the set of all conflicting claims is denoted as $\mathcal{B}$. For each conflicting claim, a rule assigns a distribution of the endowment among the players within that problem. A rule is a single-valued function $\varphi : \mathcal{B} \to \mathbb{R}_+^n$ such that $0 \leq \varphi_i(E, c) \leq c_i, \forall i \in N$ (non-negativity and claim-boundedness); and $\sum_{i=1}^{n} \varphi_i(E, c) = E$ (efficiency). Those rules used throughout various approaches are introduced below.

The proportional (P) rule suggests a distribution of the endowment proportional to the claims: for each $(E, c) \in \mathcal{B}$ and each $i \in N, P_i(E, c) \equiv \lambda c_i$, where $\lambda = \frac{E}{\sum_{i \in N} c_i}$

The constrained equal awards (CEA) rule recommends equal awards to all players, and this recommendation is subject to no one receiving more than his claim: for each $(E, c) \in \mathcal{B}$ and each $i \in N$, $CEA_i(E, c) \equiv min\{c_i, \mu\}$, where μ is such that $\sum_{i \in N} min\{c_i, \mu\} = E$

The constrained equal losses (CEL) rule results in an awards vector in which all players distribute the losses evenly, subject to no one receiving a negative amount: for each $(E, c) \in \mathcal{B}$ and each $i \in N$, $CEL_i(E, c) \equiv max\{0, c_i - \mu\}$, where μ is such that $\sum_{i \in N} max\{0, c_i - \mu\} = E$

The Talmud (T) combines the features of CEA and CEL and uses the aggregate claims' midpoint as its benchmark. Talmud suggests using the constrained equal awards rule when the available resources fall short of meeting the half-sum of the claims [115]. Otherwise, each player receives half of the claim, and the constrained equal losses rule is applied to distribute the remaining endowment: for each $(E, c) \in \mathcal{B}$, and each $i \in N, T_i(E, c) \equiv CEA_i(E, c/2)$ if $E \leq \sum_{i \in N} c_i/2$, or $T_i(E, c) \equiv c_i/2 + CEL_i(E - \sum_{i \in N} c_i/2, c/2)$, otherwise.

The random arrival (RA) rule also called contested garment [125] considers the case that each claim is fully satisfied until the endowment runs out following the order of the claimants' arrival. In order to eliminate the unfairness of the first-come-first-served scheme associated with any particular order of arrival, the rule proposes to take the average of the awards vectors calculated in this way when all orders are equally probable: for each $(E, c) \in \mathcal{B}$, and each $i \in N, RA_i(E, c) \equiv \frac{1}{|N|!} \sum_{i \in \mathbb{R}^N} min\{c_i, max\{E - \sum_{j \in N, j \prec i} c_j, 0\}\}$.

The concepts from cooperative game theory—such as the core, Shapley value, and nucleolus—differ significantly in their mathematical formulations, assumptions, and outcomes for allocating payoffs among coalition members. Each technique offers a unique perspective on how to distribute these payoffs, which can be critical for managing diverse and complex cooperative situations. For example, *Shapley value* is designed to measure the marginal contribution of each player to the coalition. *Shapley-Shubik* is similar to Shapley values, but it is particularly useful when it's important to assess how critical each member's participation is to the overall success of the coalition. The calculation considers all possible orders in which members can join, reflecting the added value each member brings when they enter the coalition. Shapley values and Shapley-Shubik solutions are ideal when the sequence of joining impacts the coalition's value, such as in sequential investment decisions or collaborative research where early contributors might bear more risk or cost. *Nucleolus* solution concept is particularly valuable when the objective is to minimize the disparity in allocations among coalition members. It seeks to find an allocation that minimizes the greatest dissatisfaction among all possible coalitions, thus ensuring a form of equity and stability that can prevent any group from feeling disproportionately disadvantaged. The nucleolus is beneficial in scenarios where fairness and the minimization of unhappiness are crucial, such as in joint ventures or collaborative projects with uneven benefits or costs. *Core* of a game includes all possible distributions of total gains among the players such that no subgroup would be better off by breaking away and forming their own coalition. This solution concept is fundamental when the focus is on ensuring that no subset of players has an incentive to defect, even if it might not address issues of equitable distribution as directly as the nucleolus. The core is particularly relevant in cooperative arrangements where the stability of the entire group is essential, like in alliances or large-scale collaborative agreements. *Banzhaf power index* does not consider the order of coalition formation, making it suitable for scenarios where the sequence of joining does not affect the coalition's value. This index is often used in voting systems to measure the power of a voter without considering the order in which votes are cast, making it applicable in settings where decisions are made simultaneously or the impact of the order is negligible. Finally, *Conflicting claims* are crucial in scenarios where members have predefined claims, and the available resources are insufficient to fully satisfy these claims. This method helps allocate limited resources in a way that attempts to consider the legitimacy of each claim as much as possible.

Each of these solution concepts can be strategically employed based on the specific needs and goals of the coalition, highlighting the versatility and depth of cooperative game theory in resolving complex allocation problems.

2.3.5 COOPERATIVE GAME THEORY EXAMPLE

This section presents a cooperative game theory example based on the Shapley value solution.

Some well-known cooperative game theory examples are matching problems, such as stable marriage [131], stable roommates [132], and the National Resident Matching Program

(NRMP) [133]. Engineering publications have widely discussed cost allocation problems, and Shapley Value was a common solution concept employed [85]. Therefore, the numeric example demonstrated below presents some of the concepts, such as characteristic function in a cost allocation problem. This example will be in the context of a jazz band game described by Malawski, Wieczorek, and Sosnowska [134].

The problem is described as follows. The club owner promises $150 to the singer, pianist, and drummer for a joint performance. N= 1, 2, 3 = singer, drummer, pianist. The characteristic function is as follows: v(1, 2, 3) =150, v (1, 2) = 60, v (1, 3) = 100, v (2, 3) = 50, v(1) = 40, v(2) = 0, v(3) = 35. For this game, we can check the superadditivity condition:

$$\begin{aligned}
v(\{1,2,3\}) &= 150 \geq 75 = v(\{1\}) + v(\{2\}) + v(\{3\}) \\
v(\{1,2,3\}) &= 150 \geq 95 = v(\{1,2\}) + v(\{3\}) \\
v(\{1,2,3\}) &= 150 \geq 100 = v(\{1,3\}) + v(\{2\}) \\
v(\{1,2,3\}) &= 150 \geq 90 = v(\{2,3\}) + v(\{1\}) \\
v(\{1,2\}) &= 60 \geq 40 = v(\{1\}) + v(\{2\}) \\
v(\{1,3\}) &= 100 \geq 75 = v(\{1\}) + v(\{3\}) \\
v(\{2,3\}) &= 50 \geq 35 = v(\{2\}) + v(\{3\})
\end{aligned}$$

Assume the following imputations for the singer (a_1), drummer (a_2), and the pianist (a_3), respectively:

$$a_1 \geq 40, \quad a_2 \geq 0, \quad a_3 \geq 35, \quad a_1+a_2 \geq 60, \quad a_1+a_3 \geq 100, \quad a_2+a_3 \geq 50, \quad a_1+a_2+a_3 = 150.$$

Given this information, we can calculate the Shapley Value. The calculation suggests the following allocation for the singer, drummer, and pianist: $\phi(i) = (67.5, 22.5, 60)$. Note that the proportional solution calculated as $\text{prop}_i = \frac{v(i)}{\sum_{i=1}^{3} v(i)} \cdot v(N), \quad i \in N$ will suggest following allocation to the singer, drummer, and the pianist $\text{prop}_i = (80, 0, 70)$. We can also calculate the core and look if the Shapley value belongs to the core. We get the following results for the core $40 \leq a_1 \leq 100, \quad 0 \leq a_2 \leq 50, \quad 35 \leq a_3 \leq 90.$ We can see that the Shapley value belongs to the core.

2.3.6 ASSUMPTIONS AND CONCERNS REGARDING GAME THEORY

Here, the concerns and assumptions regarding game theory are explored.

Game theory is a powerful technique to determine solutions for problems that are difficult to study due to the conflicting interests and strategies of the players. However, it also has some limitations.

One of the game theory-related issues discussed by many researchers is that game theory requires the assumption of rationality [135]. The concept of rationality is quite familiar to economists, but what does this mean for engineers?

In engineering, the rational choice models attempt to capture critical facets of a situation and examine engineers' decisions due to their preferences in conjunction with the constraints of a situation. The rationality of the engineers does not show their preferences over outcomes, but it describes the choices given ordinal preferences and the situation that confronts it. The rationality of the engineers does not mean that engineers will reach the same decision even when faced with the same situation. Engineers can differ not only in their choice-making process but also in their preferences over the outcomes. Rationality does not guarantee error-free decisions [136]. The reasons for undesirable consequences can be associated with information scarcity, risk, and uncertainty related to the system. The concept of rationality is essential for the decision-making process and building game-theoretic scenarios. Some researchers consider the assumption of rationality in game-theoretical models as a weakness or a limitation because people do not always act rationally. This weakness is alleviated when considering behavioral game theory. Behavioral game theory describes and analyzes decision-making using experimental data [137, 138]. It describes what people actually do. Camerer [137] states that behavioral game theory expands analytical game theory by adding emotion, mistakes, limited foresight, and learning to analytical game theory. Hence, behavioral game theory weakens the rationality assumption by employing experimental and psychological regularity.

Another limitation is related to determining the payoffs for the game, which can be challenging. For each choice of the strategies, each player receives some payoff. Quantifying the payoffs given different strategies can be complicated as well. Kreps [139] states that the payoffs are usually associated with the "pecuniary" incentives of the players. Large pecuniary payoffs are related to risk aversion, and when the outcomes are monetary, we typically assume players prefer more financial gain. Assessing the probability that the rival player will accept or reject the particular offer also requires further attention. Note that, in reality, financial gain is not the only incentive that drives people's choices and decisions. An altruistic player may care about his or her as well as other players' benefits. Hence, the payoff should represent a complete description of each player's happiness with each of the possible outcomes of the game to have a more accurate prediction of the behavior and the situation.

Other limitations of game theory are the need for precise protocols, the existence of many equilibria and no way to choose one, and even specifying the rules of the game [139].

2.4 LINEAR REGRESSION MODEL

Here, the concept of the linear regression model is described.

A linear regression model is a method used to model the relationship between a dependent variable and one or more independent variables by fitting a linear equation to observed data [140]. A linear regression model is used with cooperative game theory feature importance methods for two reasons. First, linear regression is the simplest modeling approach considered intrinsically explainable. However, some common problems in regression analysis, such as high multicollinearity, nonconstant error variance, and autocorrelated errors, affect the model prediction results. Some of these issues can be fixed by simply removing the redundant feature. In other cases, removing the feature will not be the right action. This is when explainable artificial intelligence is used to learn more about the AI system. Also, the simultaneous multiple linear regression model includes all the specified features without considering their importance values [141], and the statistical significance of these values may fail if any common problem prevails. Mathematically, the regression model is defined as follows:

$$y_i = \beta_0 + \beta_1 x_1 + ... + \beta_n x_n + \varepsilon \tag{9}$$

Where y_i are the observations of the target variable, x_1, x_2, ..., x_n are the features, ε is the regression error term, that is assumed to be normally distributed, $\varepsilon \sim \mathcal{N}(0, \sigma^2)$.

A regression model is usually evaluated based on how much error the prediction makes. Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE) are example matrices used to characterize the regression model performance. Regression outputs β_0 intercept, and β_1, β_2, ..., β_n are regression coefficients computed through the Ordinary Least Square (OLS) approach. Regression coefficients are regarded as unstandardized effect sizes as they suggest the intensity of the relationship between features and describe how important the findings are in a practical setting. For example, if the effect size is negligible, then we interpret that the variation to the feature has almost no effect on the target variable. The objective of the OLS estimator (linear regression line) is to minimize the difference between actual and fitted data points, i.e., error sum of squares (ESS). Statistically, a model fits the data well if the differences between the observed values (actual observations) and the model's predicted values are small and unbiased.

The performance of the regression model is evaluated by multiple determination R^2 coefficient. R^2 coefficient demonstrates how well the model replicates the observed outcomes.

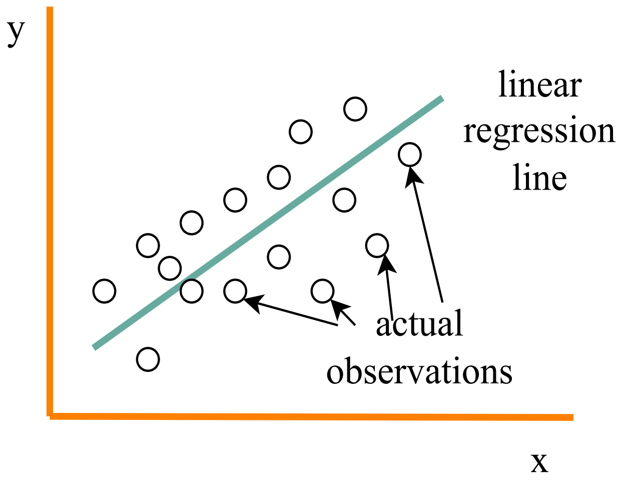


FIG. 3: Linear regression model with dots representing the observed values and linear regression line evaluated with OLS estimator

The sum of squares values is used as an indicator to present the dispersion of data and suggest how well the data fits the regression model. The three sums of squared indicators to determine the R^2 value are the total sum of squares (TSS), regression sum of squares (RSS), and error sum of squares (ESS). The formula for multiple determination R^2 coefficient is as follows:

$$R^2 = \frac{RSS}{TSS} = 1 - \frac{ESS}{TSS} = 1 - \frac{\sum(y_i - \hat{y_i})^2}{\sum(y_i - \bar{y_i})^2} \tag{10}$$

Where, y_i are the actual values, $\hat{y_i}$ denote the predicted values, and $\bar{y_i}$ is the mean of the y values.

Regression also supplies statistical significance p-values. A low p-value (< 0.05) indicates that the feature is perhaps a meaningful addition to the model. Larger p-values suggest that the feature is not appropriate for predicting the target. Regression output may be statistically insignificant with high p-values, implying some features are irrelevant to the model. However, this insignificance could be an inaccurate prediction of the system. With multiple linear regression, cooperative game theory has been employed (popular) to develop and extract new insights about the prediction that was not able to be achieved based on the regression model alone. To generate accurate predictions, the regression model follows certain assumptions, which are linearity, normality, homoscedasticity, independence, and absence of multicollinearity [142].

Linearity means that the target's mean value is a linear combination of the regression coefficients and the features.

Normality assumes that the target outcome given the features follows a normal distribution. When this assumption is violated, the estimated confidence intervals of the feature weights are invalid.

Homoscedasticity assumes that different values of the target have the same variance in their errors, regardless of the values of the features.

Independence suggests that the errors of the target variables are not correlated with each other.

Absence of multicollinearity Having strongly correlated features is problematic because it becomes hard to estimate the weights. A more detailed description of multicollinearity is described below.

2.4.1 MULTICOLLINEARITY IN REGRESSION MODELS

Multicollinearity occurs when features in a regression model are highly correlated. This violates one of the assumptions in a regression model, i.e., the features should be independent. Violating multicollinearity may not impact the prediction but can impact inference. P-values typically become statistically insignificant even though the feature may be essential for the prediction. Variance Inflation Factor (VIF) is used to measure the severity of multicollinearity in regression analysis. VIF shows the increase in the variance of a regression coefficient as a result of collinearity [143]. Computationally, it is defined as the reciprocal of tolerance: $\frac{1}{1-R^2}$. Lower levels of VIF are desired, as higher levels of VIF are known to affect adversely the results associated with a multiple regression analysis. The correlation matrix is another way to show the correlations between the features. One way to deal with multicollinearity is by using principal components analysis (PCA) [144], or partial least square regression (PLS) instead of OLS regression [145]. PLS regression can reduce the features to a smaller set with no correlation among them. In PCA, new uncorrelated variables are created. It minimizes information loss and improves the predictability of a model. These solutions to address the multicollinearity issue suffer from limitations, including a major limitation for PLS to overlook, including real correlations and sensitivity to the relative scaling of the descriptor variables. A major disadvantage to using PCA is the difficulty in interpreting the data and identifying which are the most important features in the model after computing principal components. I believe explainable artificial intelligence can be useful in addressing the multicollinearity issue and provide the needed clarification about the regression model predictions.

2.4.2 LINEAR REGRESSION MODEL AND EXPLAINABLE ARTIFICIAL INTELLIGENCE

Letzgus [146] discuss XAI with the regression model, suggesting that little attention has been devoted to XAI for regression models (XAIR). The work presents the conceptual differences of XAI for regression and classification tasks and establishes novel theoretical insights and analysis for XAIR. Explanation methods that are based on Shapley values are particularly favorable in the regression scenario. This is because Shapley values allow for a decomposition of the predicted quantity on the input features that saves the explanation in the same measurement units as the prediction tasks.

Lipovetskey and Conklin [37] present the Shapley net effect technique that uses Shapley values for the regression model to measure the feature importance values [37]. The method is a supervised learning approach designed to estimate the marginal contributions and relative importance of the highly correlated features in regression models. Feature importance evaluation consists of comparing the model performance, measuring the multiple determination R^2 value, with and without particular feature $\mathtt{i}$ using Shapley value We will get Eq. 5 to measure the features' relative importance for the estimated model:

$$U_i = R^2 - R_i^2 \tag{11}$$

Subsection 3.4.1 presents a detailed description of the algorithm and the analysis necessary to measure the feature importance values using Shapley values.

2.5 LOGISTIC REGRESSION MODEL

Here, the concept of the logistic regression model is described.

Logistic regression is a statistical method used for binary classification, which involves predicting the probability of an observation belonging to one of two classes [34]. Logistic regression is widely employed in various fields, such as medicine [147] and finance [148]. The primary purpose of logistic regression is to model the relationship between a binary dependent variable and one or more independent features, providing a probabilistic estimate of the likelihood of an event occurring [34].

The logistic regression equation is derived from the logistic function, also known as the sigmoid function. The logistic regression model transforms a linear combination of input features using the sigmoid function, constraining the output to a range between 0 and 1. The equation can be expressed as:

$$P(Y = 1|\mathbf{x}) = \frac{1}{1 + e^{-(\beta_0+\beta_1 x_1+\beta_2 x_2+\ldots+\beta_p x_p)}} \tag{12}$$

Here, $P(Y = 1)$ is the probability of the dependent variable Y being equal to 1, e is the base of the natural logarithm, β_0 is the intercept, $\beta_1, \beta_2, \ldots, \beta_n$ are the coefficients associated with the independent variables $X_1, X_2, \ldots, X_n$. The output of logistic regression provides predicted probabilities, and a decision threshold is chosen to classify instances into the two categories.

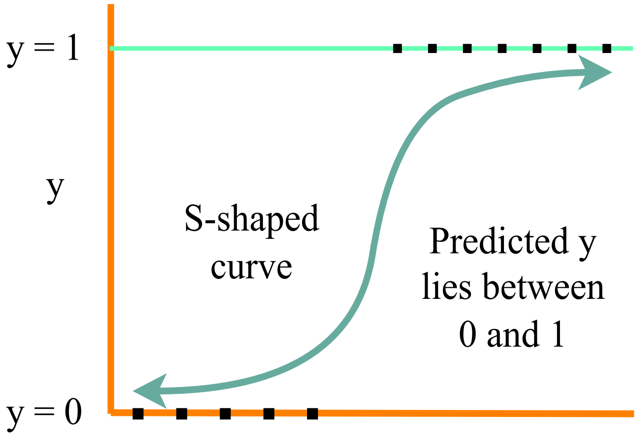


FIG. 4: Logistic regression

The model estimates the odds of the event occurring, and the log-odds ratio is used to make predictions. The coefficients β indicate the direction and strength of the relationship between each independent variable and the log odds of the event. Logistic regression assumes a linear relationship between the log odds and the features.

Properties of the logistic regression equation include:

- Target y obeys Bernoulli distribution,
- Prediction is based on the maximum likelihood estimator.

Logistic regression is not only valuable for classification but also for understanding the influence of features on the target outcome, making it a versatile tool in the realm of machine learning modeling [59].

Assumptions for implementing logistic regression are [149]:

1. Target is binary or dichotomous.
2. Little or no multicollinearity between features and target.
3. Linear relationship of features to log odds.

4. Large sample size.

5. No extreme outliers.

6. Independent observations.

As indicated in Section 2.4, linear regression often relies on R^2 to gauge the goodness of fit, and logistic regression uses alternative metrics like concordance to evaluate the model's performance in capturing the underlying patterns in binary classification problems. However, logistic regression can provide pseudo R^2 values, such as McFadden's R^2, which offer a similar concept as R^2 in linear regression, but adapted for classification tasks [150, 151]. McFadden's R^2 is defined as:

$$R^2_{\text{McFadden}} = 1 - \frac{\text{Log-Likelihood of the Null Model}}{\text{Log-Likelihood of the Model}}$$

The Log-Likelihood of the Model is the logarithm of the likelihood function for the logistic regression model, and the Log-Likelihood of the Null Model is the logarithm of the likelihood function for a model with no predictors (i.e., only the intercept). R^2_{McFadden} ranges from 0 to 1, and a higher value indicates a better fit. Values close to 0 suggest that the model doesn't improve much over a null model (a model with no predictors). In addition, other performance evaluation metrics, such as F1 score, precision, recall, and accuracy, could be applied to assess the effectiveness of the model in classification tasks.

2.5.1 LOGISTIC REGRESSION MODEL AND EXPLAINABLE ARTIFICIAL INTELLIGENCE

Logistic regression is widely used with explainable AI techniques to enhance the transparency and interpretability of its predictions [13, 152]. When combined with XAI, it becomes easier to understand how specific features contribute to the model's decision-making process for classification tasks. Logistic regression coefficients indicate the contribution of each feature to the prediction. These coefficients can be interpreted directly, providing a clear understanding of the role of each feature in the model [153]. Features with larger coefficients have a more significant impact on the predicted outcome [153]. Logistic regression is usually considered a simple interpretable model [22, 5]. However, this model may struggle to capture complex interactions between features, impacting its ability to provide accurate explanations for certain predictions [154]. LIME presented in Subsection 2.2.1 is an example of a widely used explainable AI technique that uses logistic regression to provide explanations to classification tasks. A specific example of a logistic regression model could involve predicting the probability of a customer making a purchase, considering variables such as age, income, and browsing history. When applying LIME, a local explanation may be generated for a specific customer, highlighting the significance of their recent browsing history in shaping the purchase decision.

2.6 AGENT-BASED MODELING

An agent-based model (ABM) is a type of computational model that simulates the behavior of individual agents and their interactions with each other and their environment [38]. These agents may either be identical or possess distinct characteristics [155]. The number of agents can vary from just one to potentially reaching into the millions. Since these models operate based on computations, the behavior of the agents is governed by rules [156]. Rules can range from being simple, like employing backward induction on an extensive game form, to being more intricate, such as being derived from heuristics rooted in cognitive psychology or neuroscience. The model is designed to track the interactions between agents' behavior over time.

In an ABM, agents interact with their environment by receiving inputs and responding with actions. Consequently, numerous ABMs incorporate elements like social networks or spatial relationships to influence decision-making. When the ABM software simulates the collective behaviors of these individual agents, it gives rise to system-level outcomes such as cooperation and fluctuations. These outcomes are often described as emergent, generative

[156], or originating from the bottom-up [157].

The outcomes in ABMs can be temporal [158], stochastic [159], and static equilibria, such as in Schelling's racial segregation model [160] or Axelrod's culture model [161]. In contrast to many modeling approaches commonly employed in the social sciences, ABMs frequently place their emphasis on understanding the dynamic aspects of the behavior being studied. In these models, finding equilibria can often prove challenging or may not even be feasible due to the complex and non-linear nature of the systems under investigation [162]. Nevertheless, even in situations where traditional equilibria are elusive, discernible patterns of behavior can still emerge within ABMs [163]. This adds a layer of complexity to ABMs [163]. Complexity refers to systems and processes that are difficult to explain, predict, or engineer [164, 165, 166, 167].

ABMs are used in a variety of fields, including economics and finance [168], ecology [169], sociology [170], and epidemiology [171], to study complex systems and understand the emergent behaviors that arise from interactions between individual components. The study of an agent's cooperative behavior explores how individual agents decide to collaborate, form partnerships, or engage in collective actions to achieve shared objectives. Cooperative behavior involves agents working together in a way that is mutually beneficial. Modeling cooperative behavior among agents can be complex, as it involves considering factors like trust, information sharing, negotiation, and the division of benefits. Researchers may consider human subject experiments and use mathematical models, simulation tools, and computational experiments to explore the outcomes of cooperative interactions, such as the allocation of resources, collective decision-making, and the emergence of cooperative equilibria [100, 172, 101].

Note that cooperative game theory has previously been applied to agent-based modeling to study coalition formation behavior [101, 172, 100].

A classic example of an ABM is the predator-prey model, which simulates the interactions between populations of predators and prey in an ecosystem. In this model, agents are divided into two categories: predators and prey [173]. Each predator agent is programmed to seek out and consume prey agents, while each prey agent is programmed to avoid predators and reproduce. The Lotka-Volterra equations, also referred to as the predator-prey equations, are a pair of differential equations that capture the dynamics of interacting populations [174]. These equations enable the model to track the populations of predators and prey over time, unveiling emergent behaviors such as population cycles and extinction events as a result of the interactions between agents. Specifically, the equations are given by:

The Lotka-Volterra equations, also known as the predator-prey equations, can be represented as a single equation:

$$\begin{aligned} \frac{dx}{dt} &= \alpha x - \beta xy, \\ \frac{dy}{dt} &= \delta xy - \gamma y. \end{aligned} \tag{13}$$

In Equation 13, x represents the population of the prey species, y represents the population of the predator species, and α, β, γ, and δ are positive constants representing the growth rates and interaction strengths between the populations. A predator-prey model is a useful model for studying the dynamics of ecosystems and understanding the impacts of environmental factors on populations of animals.

An example of a predator-prey model from agent-based modeling is the ”wolf-sheep predation model,” which simulates the interactions between wolves and sheep in a given ecosystem [175]. In this model, agents representing wolves and sheep move around the environment and interact based on specific rules, such as the wolves hunting the sheep and the sheep trying to avoid them. Another example that I have considered in this study is the rotifer-algae predator-prey model [176]. Rotifers and algae are common microorganisms found in aquatic environments. Rotifers typically have a transparent, elongated body with a distinctive head crowned by cilia. In Figure 5, the rotifer is described with yellowish/orange pigment. The cilia create a rotating motion, resembling a spinning wheel, as they move through water. Algae, on the other hand, encompass a diverse group of photosynthetic organisms ranging from microscopic single-celled forms to large, multicellular seaweeds. Algae can exhibit a wide array of colors, including green, red, and brown. They play a critical role as primary producers, converting sunlight, water, and carbon dioxide into organic compounds through photosynthesis. Furthermore, algae serve as a fundamental food source for numerous aquatic organisms. In Figure 5, the depicted algae are prominently green in color.

The interaction between these two species is of ecological importance, as it affects the balance of the ecosystem. In the rotifer-algae model, the agents are assumed to interact with each other based on simple conditions such as feeding and reproducing, and their individual behaviors can lead to the emergence of complex patterns at the system level. In the rotifer-algae model, each rotifer and algae agent has its own set of attributes and behaviors, such as movement, feeding, and reproduction. Rotifer-algae system incorporates various environmental factors such as light, temperature, and nutrient availability. By simulating the behavior of individual agents, the model can predict the emergent effects of adaptive behavior and the impact of different conditions on the system as a whole.

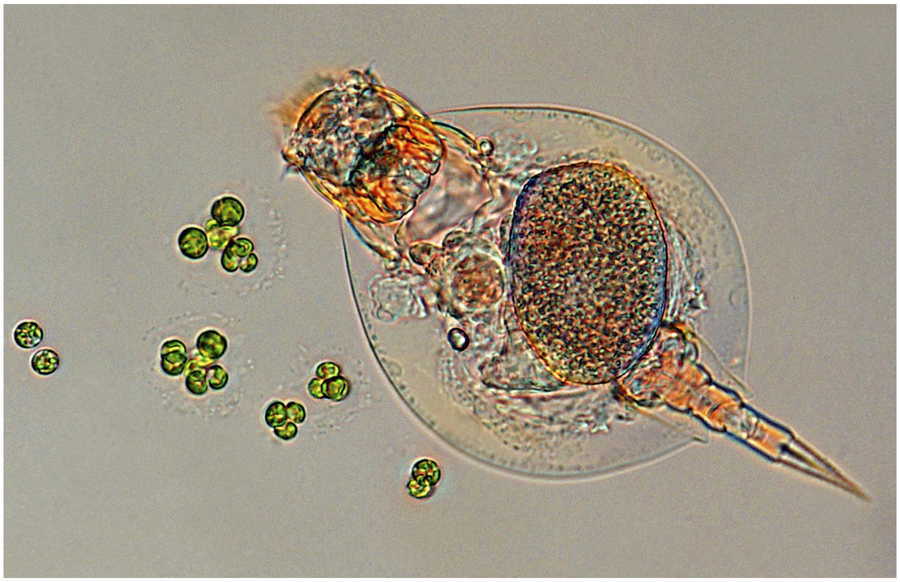


FIG. 5: Rotifer-algae predator-prey system. Image used with permission from SciencePhoto - Image ID C025/3764, Request ID 904451.

The agents interact with each other based on their proximity and the conditions governing their behavior (Figure 5). For example, a rotifer may move towards an algae if it is hungry and within a certain distance and then feed on the algae if it is close enough. Similarly, algae may reproduce when it has enough nutrients and space.

2.6.1 AGENT-BASED MODELING AND EXPLAINABLE ARTIFICIAL INTELLIGENCE

This section is adapted from a paper developed in the scope of this dissertation and published in Winter Simulation Conference [177].

A single simulation run can provide valuable insights but cannot account for all the sources of uncertainty and variability in the modeled system [178]. Therefore, multiple simulation runs with varying input parameters, initial conditions, or model assumptions are necessary to explore a wide range of possible outcomes, especially for agent-based models that simulate individual agent behavior and interactions within a larger system. However, multiple simulation runs can also introduce uncertainty when different results are observed, making it difficult to discern the significance of each input variable in the agent-based model. Given that agent-based models are inherently stochastic and sensitive to small changes, multiple simulation runs are crucial to fully explore the range of possible outcomes and evaluate the uncertainty associated with the model results [179, 101]. Although running multiple simulations can help identify aggregate patterns and emergent behaviors not apparent in single simulations, it can be computationally expensive and time-consuming. The sheer volume of data generated from multiple simulation runs can also make it challenging to identify and interpret the most significant results [180, 181, 182, 183, 184, 185].

Running multiple simulations or experiments with varying conditions or parameters can produce different results, making it challenging to determine the most accurate result [186]. Uncertainty quantification (UQ) plays a significant role in addressing this challenge by identifying, quantifying, and reducing uncertainties associated with models, algorithms, and predicted quantities of interest [187]. UQ is particularly important when accurate predictions or decisions are required, but underlying models or data are incomplete, imperfect, or subject to variability. As Begoli et al. [188] note, predictions without UQ are neither predictions nor actionable.

Addressing uncertainty in agent-based models poses several challenges. One of the main obstacles is the complexity of these models, which often have high-dimensional feature spaces with numerous parameters, initial conditions, and rules governing agent behavior. This complexity makes it challenging to identify the sources of uncertainty and the impact of each parameter on the model output. Furthermore, the stochastic nature of ABMs can pose another challenge by leading to high variability in the model output. Running multiple simulations considering different conditions can help capture the collective trends and novel phenomena that arise from agent interactions and are not observable in single simulations [189, 190]. ABMs often rely on incomplete or imperfect empirical data, leading to additional uncertainty [191]. The model's assumptions about the behavior and interactions of agents may also not accurately reflect the real-world system being modeled, adding to the uncertainty.

Sensitivity analysis has been widely used to address the challenges associated with uncertainty in ABMs [192]. However, the objective of sensitivity analysis is to improve the robustness of a model by examining how changes in the inputs of a model affect its outputs [40]. While sensitivity analysis can identify the most influential parameters and assumptions, it may not provide additional insights into the underlying mechanisms driving the model output [39].

Feature importance techniques from explainable artificial intelligence (XAI), on the other hand, could be useful in addressing these limitations. By providing additional insights into the relative importance of model features, XAI methods can enhance the initial information and help clarify and better explain the model [50]. This can lead to a more comprehensive understanding of the model's behavior and improve the accuracy and reliability of its predictions.

A paper [177] developed in the scope of this dissertation demonstrates the use of feature importance measures from explainable AI as a means for uncertainty quantification of input data that can be used when ABM simulations are designed. To achieve this, a classical predator-prey model involving two interacting species was considered: a predator (rotifer) and a prey (unicellular algae). When features are uncertain or poorly characterized, the output of the model may also be uncertain or unreliable. By quantifying the impact of these important features on the output, we can better understand the sources of uncertainty and develop strategies to reduce it. Therefore, feature importance analysis could be an essential tool in the UQ process, allowing for a more comprehensive assessment of the reliability, accuracy, and explainability of agent-based models. Measuring the importance of features in a model can be accomplished using various approaches, including permutation feature importance [82], and cooperative game theory-based solutions [193]. Cooperative game theory-based approaches are known for their ability to yield fair assessments of feature importance values [15]. By ranking the features by importance, we can focus on the most important features when making decisions about how to reduce uncertainty and improve the model's reliability and accuracy.

CHAPTER 3

METHODOLOGY

This chapter describes the methodology and experimental design of this dissertation. It outlines the components involved, such as data, models, the new explainable artificial intelligence methods derived from cooperative game theory solutions, and the weighted Shannon entropy-based permutation importance evaluation (PRIME) metric, another contribution of this dissertation. Emphasis is placed on the feature importance methods' algorithmic foundations and the execution of the PRIME metric.

Three experiments were conducted to assess these feature importance methods. The first experiment employs a linear regression model using the Searpos dataset [194] characterized by significant multicollinearity among its features. It includes conducting 30 permutations as part of the application of the PRIME metric. The second experiment applies a logistic regression model to the Adult Income dataset [195], which contains independent features. This second experiment includes 60 permutations for the PRIME evaluation. Lastly, the third experiment employs empirical data describing the predator-prey scenario [176] to measure the feature importance values and analyze the effects of input and parameter modifications for agent-based models.

For each experiment conducted, the weighted Shannon entropy-based permutation importance evaluation (PRIME) metric was applied to assess the consistency associated with the importance values of the features identified by various methods. PRIME involves two existing methods - permutation tests [82, 196] and weighted Shannon entropy [197, 198, 199, 200, 201]. In the scope of the PRIME, different number permutations (p) were evaluated: $p \in \{10, 20, 30, 50, 60, 100\}$. These permutations involve random shuffling observations within the dataset and assessing the variation in feature importance values, along with their respective rankings [82, 196]. The results indicated that even a limited number of permutations (e.g., 20) produce feature importance rankings closely resembling those generated with a larger number of permutations. Feature importance ranking refers to the process of ordering features according to their importance values [202, 203]. These rankings list the features based on how they contribute to the model's ability to make predictions. Rankings could help in identifying the most influential factors [204], simplifying model interpretation, and guiding the selection of important features [205].

In PRIME, Weighted Shannon Entropy, an advanced concept derived from information theory [198, 200, 201], has been used with permutation importance tests. This integration is designed to quantify the uncertainties associated with feature importance values generated by various feature importance methods following permutations. PRIME's objective is to observe the permutations' impact on the method's consistency in measuring the importance values of the features. Overall, PRIME is designed to better understand the stability and sensitivity of feature importance scores assigned by the feature importance methods. Also, PRIME could be used to compare various feature importance methods. Additionally, PRIME enables the direct comparison of different feature importance methods, enhancing the ability to discern the most effective methods for identifying the most important features in the machine learning model.

The experiments have been executed using R version 4.1.2 on a Microsoft Windows 10 Pro, version 10.0.19043. To measure the feature importance values with game theory solutions, the GameTheory package was used, available in Comprehensive R Archive Network at `http://CRAN.R-project.org/package=GameTheory`. The GameTheory package depends on lpSolveAPI to perform linear programming optimization. Data and the analysis are available as a Python Jupyter Notebook file online from `https://github.com/grigoryangayane/XAI_CGT_Methods`. Together, these experiments are intended to gain insights into the application of different feature importance methods developed in the scope of this dissertation.

Figure 6 outlines the methodology overview, describing the general procedure and steps this dissertation follows to address the research questions described in the Introduction. The methodology consists of a review of the methods, data, and models, method development, and evaluation.

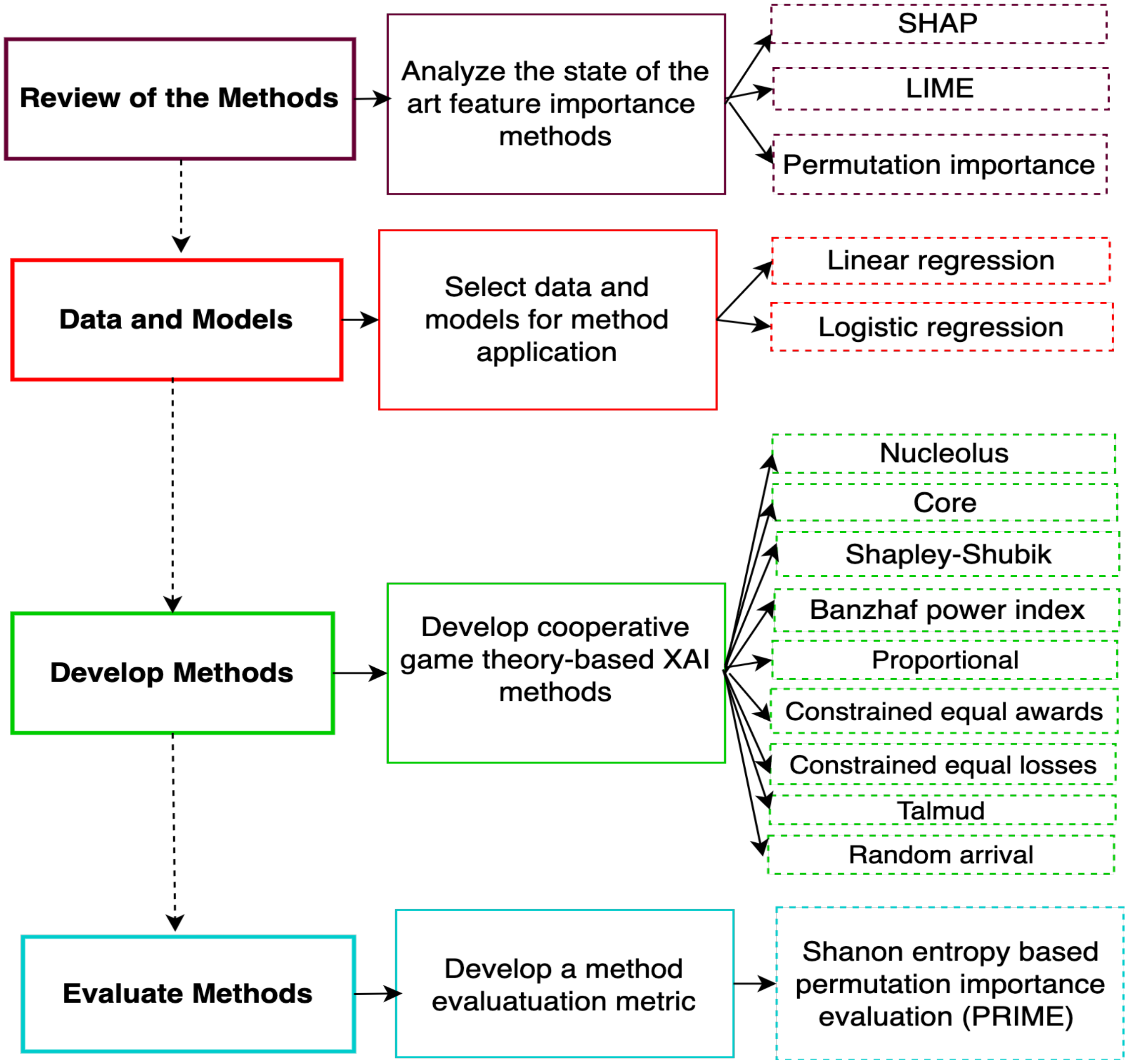


FIG. 6: Research methodology overview

1. *Review of methods:* The methodology begins with a review of existing methods, and three widely used methods (Shapley additive explanations (SHAP), local interpretable model agnostic explanations (LIME), and permutation importance) are presented in Section 3.1. SHAP leverages the Shapley value from cooperative game theory to fairly assign importance values to features according to their contribution to model predic-

tions [15]. LIME proposes employing surrogate models (more explainable models) to simplify and explain complex model predictions [13]. Permutation importance methods assess feature significance by observing changes in model performance when feature values are randomly shuffled, providing a direct empirical evaluation of each feature's impact [82].

2. *Data and models:* Following this review of the existing methods, in Section 3.2 and 3.3, the methodology progresses to the selection and preparation of the data, detailing the sources and description of the data for the analysis. This data refers to the initial datasets Seatpos, Adult Income, and Predator-prey, which are used for the first phase of the feature importance methods to develop the explainable models. In this dissertation, linear and logistic regression models were developed. These models were used due to their transparency and straightforward interpretation [206, 22]. Afterward, from these developed models, performance metrics such as the R-squared value are extracted. These R-squared values then form a second set of input data, which is employed in the final phase of feature importance methods to assess the impact of each feature.

3. *Develop methods:* Building on this foundation, the methodology introduces the development of new feature importance methods in Section 3.4. This phase describes the algorithms and the steps of the new methods, designed to extract explanations about the feature importance values. While the discussion of algorithms references regression models, it is important to note that any model capable of generating performance metrics could be applicable. In the algorithms, the model type and performance metrics can be updated to facilitate feature importance analysis with a different model.

   Figure 7 presents the sequence of steps based on the data, models, and the feature importance method development discussed above.

   Figure 7 illustrates the development process of feature importance methods, beginning with data and model construction using this data. Subsequently, performance indica-

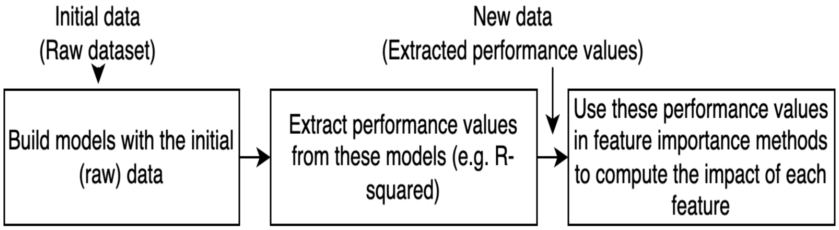

Initial data
(Raw dataset)
New data
(Extracted performance values)
Build models with the initial (raw) data
Extract performance values from these models (e.g. R-squared)
Use these performance values in feature importance methods to compute the impact of each feature

FIG. 7: Process diagram of feature importance method execution

tors are extracted from these models. These performance indicators then serve as the new input for the feature importance methods. The description and the algorithmic steps of the feature importance methods are presented in Section 3.4.

4. *Evaluate methods:* Finally, the methodology culminates in the evaluation of these newly developed methods. This evaluation is conducted through the Weighted Shannon entropy-based permutation importance evaluation (PRIME) metric, which is an integration of two existing methods, weighted Shannon entropy [198, 200, 201] and permutation test [82, 196], along with a series of experiments designed to test the methods' consistency and uncertainty in feature importance rankings. Section 3.5 describes the PRIME evaluation method.

Below, each component of this methodology is presented in detail, as well as the experimental design describing the methodology implementation.

3.1 REVIEW METHODS

The first step of the methodology consists of reviewing the literature on the state-of-the-art features importance and feature selection methods, to understand the landscape of existing methods, their applications, and objectives. The review aimed to encompass widely used approaches covering various domains. The search strategy involved accessing academic databases such as IEEE Xplore, and Google Scholar, utilizing relevant keywords such as ”feature importance,” ”feature selection,” and ”variable importance.”

The selected studies were analyzed to extract information regarding the methodologies employed. Special attention was given to recent advancements and emerging trends in feature importance, ensuring a comprehensive understanding of the current state of the art. The state-of-the-art methods that were reviewed are:

1. Shapley additive explanations (SHAP)

2. Local interpretable model-agnostic explanations (LIME)

3. Permutation importance

Detailed descriptions of these methods are presented in Chapter 2 Section 2.2. These reviewed methods represent widely adopted and established approaches across diverse fields. Additionally, these methods were implemented on a selected dataset to evaluate the importance and impact of different features in the prediction.

Simultaneously, the analyzed feature importance methods play a pivotal role in assessing the significance and impact of individual features on the prediction. By quantifying the contribution of each feature to the overall model performance, they offer valuable insights into the factors driving predictive outcomes. These methods (SHAP, LIME, and permutation importance) often adapt to various machine learning algorithms and dataset types, enhancing their applicability across various research domains [207, 208, 203]. These methods are applied to one of the datasets (Seatops), and the results are presented in Chapter 4, Section 4.1.

3.2 DATA

This section describes the datasets used for the development of feature importance methods. To evaluate the explainable artificial intelligence (XAI) techniques developed in this dissertation, a diverse set of datasets was selected, encompassing continuous data (Seatpos), categorical data (Adult), and time series data (Predator-prey). These datasets were chosen to provide a comprehensive assessment across various domains, including transportation (Seatpos), socio-economics (Adult), and biology (Predator-prey), ensuring wide applicability of the feature importance methods developed. The selection was made to cover different data types (continuous, categorical, time-series), models (linear, logistic, and agent-based models), and domain challenges, illustrating the versatility and effectiveness of the XAI techniques in addressing diverse feature importance needs. Each dataset is described in detail below.

Seatpos dataset [194] is employed to evaluate the XAI methods based on a regression model. This data was collected by HuMoSim laboratory researchers at the University of Michigan. The dataset is designed to analyze car seat positions based on the demographic attributes of 38 drivers, encompassing a total of 38 observations (rows). The features included in the dataset are numeric and are used to model the car seat position. The dataset includes features for demographic and physical measurements: Age (age in years), Weight (weight in lbs), HtShoes (height with shoes in cm), Ht (height barefoot in cm), Seated (seated height in cm), Arm (lower arm length in cm), Thigh (thigh length in cm), Leg (lower leg length in cm), and hipcenter (horizontal distance of the hips' midpoint from a fixed car location in mm). This data is important as interior design has been linked to traffic accidents in previous studies [209]. To address this issue, various measures have been taken to establish better car designs. Knowing the dimensions of the driver helps the manufacturer in designing a car seat that provides the maximum possible safety. A regression model is used for the prediction, and the hipcenter is the target variable and proxy measurement for a car seat, and the rest of the variables are the features to explain the hipcenter.

This dataset was chosen specifically because of its significant multicollinearity among the features. The goal was to assess how effectively the newly developed feature importance methods can predict the importance values of features in an interconnected setting.

Adult Income dataset from the 1994 US Census Bureau database is considered [195] and applied to a logistic regression model. The dataset analyzed contains 48,842 observations (rows) in total, and the data type is categorical and integer. This dataset analyzed contains 11 features (columns); however, nine features were used for the analysis after removing certain features that are not very informative, such as an individual's identification number (ID) and reference numbers. The Adult dataset pertains to socioeconomic classification based on demographic and socioeconomic information, and the prediction task is to determine whether a person makes over $\$50,000$ a year. The features within the dataset describe factors associated with individuals' annual income, such as the individuals' education level, age, sex, occupation, marital status, work class, race, hours worked per week (hours_per_week), and native country. Categorical variables include workclass (types of employment with some missing values), education (levels of education), marital-status (marital conditions), occupation (job types), relationship (family relations), and race (racial categories), and native country (country of origin), all without missing values except for workclass and occupation. Additionally, there's a binary variable for sex (Female, Male). Age is a numerical variable. These are the features used in the logistic regression model to estimate the individual's income level: high if it exceeded $50,000\$$ and low otherwise.

Predator-prey dataset collected by Blasius et al., [176] was used to analyze agent-based modeling system. The dataset comprises time series data from ten physical experiments involving a planktonic predator-prey system, with measured population densities of the prey (unicellular algae), predator (rotifer), and predator life stage characteristics recorded over approximately 2,000 measurement days (rows, observations) and 6 features (columns). These features include the total number of rotifers, unicellular green algae, produced eggs, dead animals, egg ratio, and external factors, such as spatial structure, immigration, or environmental perturbations, to investigate the potential for persistent cycles. The dataset demonstrates predator-prey cycles of unparalleled length, making it a valuable resource for investigating the dynamics of predator-prey systems.

Overall, the data can encompass continuous values, temporal data, or text-based information depending on the specific context and application.

3.3 MODEL

This section outlines the models employed in the development of the new feature importance methods, specifically highlighting the inclusion of linear regression and logistic regression models. Linear regression is crucial for understanding relationships between continuous variables, allowing us to quantify the impact of each variable on a continuous outcome. On the other hand, logistic regression is essential for analyzing binary outcomes, providing insights into how different variables influence the probability of a particular event or classification.

Linear regression: For the regression model, data is necessary to predict the relationship between features and the target [210]. The output for a regression model that will be extracted is the regression R^2 coefficient. R^2 coefficient demonstrates how well the model replicates the observed outcomes. The sum of squares values is used as an indicator to measure the dispersion of data and suggest how well the data fits the regression model. The three sums of squared indicators to determine the R^2 value are the total sum of squares (TSS), regression sum of squares (RSS), and error sum of squares (ESS). Equation 10 is used to measure the R^2 values. A more detailed description of the linear regression model is described in Section 2.4.

Logistic regression, described in Section 2.5, is used for modeling the probability of a binary outcome. In other words, it predicts the probability that an instance belongs to a particular category [34]. Despite the name ”regression,” logistic regression is a classification algorithm. It is widely used when the dependent variable is categorical and binary, meaning it has only two possible outcomes (e.g., 0 or 1, Yes or No, True or False). The logistic regression model is based on the logistic function (also known as the sigmoid function), which transforms any real-valued number into a value between 0 and 1. The model calculates the odds of the event happening and then transforms these odds using the logistic function to provide a probability. The McFadden R-squared is a metric used to assess the goodness of fit in logistic regression models [150, 151]. McFadden R-squared is mainly used for comparing model performances. The description of the logistic regression model, as well as the McFadden R-squared value, is provided in Chapter 2, Section 2.5.

3.4 EXPLAINABLE ARTIFICIAL INTELLIGENCE METHODS

This section describes various cooperative game theory-based feature importance methods developed in the scope of this dissertation.

To develop feature importance methods using game theory, it is essential to conceptualize the problem as a cooperative game. Defining feature importance as a cooperative game implies that features can collectively influence the model's predictive power. Each feature can be seen as a "player" in the cooperative game, and the interactions between features are considered cooperative rather than competitive. Features may have dependencies, and their combined impact on the model's performance can be greater than the sum of individual contributions. By approaching feature importance through the lens of a cooperative game, the method recognizes the potential for collaborative influence and importance beyond individual feature effects. This involves understanding how features jointly contribute to achieving the final outcomes.

Before delving deeper into the cooperative game theory-based feature importance solutions, let's understand what a feature contribution is. Non-mathematically, the feature contribution can be described as the difference that a particular feature brings to the final performance of the prediction. For example, the Shapley value considers the difference when having a particular feature in the prediction analysis compared to when it is not included. Note that the feature can be used alone to conduct the prediction analysis (univariate analysis) or in combination with other features (multivariate analysis). Thus, intuitively, the final contribution of the feature should be a form of the average of all the possible combinations of models.

The goal of integrating cooperative game theory solutions into explainable artificial

TABLE 2: Game theory solution categorization

Game class	Game theory solution
Transferable utility games	Shapley value
	Core
	Nucleolus
Voting games	Shapley-Shubik index
	Banzhaf power index
Conflicting claims games	Proportional
	Constrained equal awards (CEA)
	Constrained equal losses (CEL)
	Talmud (T)
	Random arrival (RA)

intelligence for measuring feature importance is to achieve a fair distribution of these values, reflecting the unique impact of each feature. Cooperative game theory has been instrumental within the field of XAI, providing insights into the opaque mechanisms of black-box machine learning models [15, 14, 211, 37]. Particularly, the concept of Shapley values, a solution from cooperative game theory, has gained widespread adoption. The objective of Shapley value based explainable artificial intelligence methods is to provide a more transparent and understandable explanation of how different features impact the model's output [15, 14, 211, 37]. However, the Shapley value-based feature importance methods suffer with several limitations which are discussed in Section 3.4.1.

Table 2 describes the game theory methods used to compute the feature importance values. The description of these methods is presented in Subsections 2.3.2 (transferable utility games), and 2.3.3 (voting games), 2.3.4 (conflicting claims). The next subsection presents an overview of these methods that are used for assessing the importance values of the features in machine learning models. These methods utilize explainable techniques, such as linear or logistic regression, as surrogate methods to explain black box machine learning models. Surrogate models refer to simplified models that are used to approximate the behavior of a more complex, black-box model [13]. The primary purpose of surrogate models is to provide a more interpretable understanding of the complex model's decision-making process.

Figure 8 shows the connection between the black box and explainable surrogate models (Defined in Chapter 2). Black box models are more complex and may lack interpretability [212, 4]. Examples include deep neural networks and ensemble models [4]. The explainable surrogate model is a simplified and interpretable model created to approximate the behavior of the black-box model [13]. Common explainable models include linear and logistic regression and decision trees [22]. Empirical data or data generated by a black box model is used as input data for the explainable surrogate model. This surrogate model is trained to explain the prediction of the black-box model [13]. Training refers to the process of developing the explainable surrogate model to approximate or explain the predictions made by the black box model [13]. This involves using the input data that was fed into the black box model, along with its predictions, and developing a surrogate model to explain the model predictions in a more interpretable manner. The agent-based model experiment outlined in Section 4.4 adheres to the same rationale, treating empirical data as input to describe the black-box agent-based model. Subsequently, it constructs an explainable model utilizing linear regression to generate insights into the workings of the agent-based model.

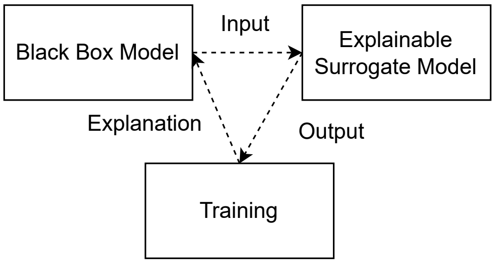


FIG. 8: Connection between a black-box model and an explainable surrogate model

The surrogate model serves as a tool for understanding and explaining the relationships between inputs and outputs in a more transparent manner by conducting some training or analysis [213]. The methods developed in the scope of this dissertation are presented in Table 3.

Subsequent subsections discuss each method in detail, considering regression models

TABLE 3: Cooperative game theory based feature importance methods developed in the scope of this dissertation

Method	Acronym
Shapley feature importance	SFI
Core Feature Importance	CorFI
NuCleolus Feature Importance	NcFI
SHapley-SHubik Feature Importance	SH2FI
Banzhaf-power Feature Importance	BFI
Constrained Proportional feature importance	CPI
Constrained EQual Awards feature importance	CEqA
Constrained EQual Losses feature importance	CEqL
Conflicting Claims Talmud Valuation	CCTV
Conflicting Claims Random Arrival	CCRA

and their corresponding R-squared values. These methods are designed to be model-agnostic, implying that they are compatible with any model. The algorithms outlined in these methods reference regression models for ease of explanation. However, the regression models can be substituted with other explainable models that suit the data and the specific problem of interest. The cooperative game theory-based feature importance methods using linear regression as a surrogate model are further examined in the context of an agent-based model (Section 3.4.4). The feature importance method based on linear regression was employed to gain insights into the empirical data describing the predator-prey agent-based model. This method, utilizing a linear regression model as a surrogate, is aimed to facilitate an understanding of the impact of input and parameter changes on the feature importance values that could be used in the simulation design and development process.

3.4.1 TRANSFERABLE UTILITY-BASED FEATURE IMPORTANCE

This section presents transferable utility (TU) methods, such as the Shapley Value, the Core, and the Nucleolus. These methods could be used in machine learning to understand and interpret the contribution of each feature in a model. The description of these methods is presented in Section 2.3.2. These methods are grounded in the concept of transferable utility, where the value generated by a coalition of players (or features in the context of machine learning) can be distributed among these coalition members. The aim is to quantify how much each feature contributes to the predictive power or the performance of the model. For instance, the Shapley Value feature importance (SFI), with its emphasis on an individual's marginal contribution to a coalition, can be adapted to measure the incremental impact of each feature on the model's performance, thereby offering a fair and comprehensive assessment of feature importance. The core feature importance (CorFI) can be used to identify sets of features that collectively contribute to model robustness, ensuring that no subset of features would provide a better prediction if used alone. The Nucleolus (NcFI), focusing on minimizing dissatisfaction (or prediction error in this context), can help in optimizing the combination of variables to achieve the most stable and accurate model outcomes. These TU solution concepts thus could help better understand the relevance of features in black-box machine learning models, enhancing the explainability of these models.

Notice that Shapley values have been extensively used to measure feature importance values [15, 14, 37]; however, it has been criticized due to many limitations, such as mathematical and human-centric issues, the additivity constraint associated with Shapley-value-based explanations [26, 25, 1] or the challenge to accurately identify feature importance values when features are highly correlated [1]. Core and Nucleolus have been considered in the following study by Yan and Procaccia [21]. This study considers Monte Carlo sampling and a logistic regression model to conduct feature importance analysis and shows that achieving results for the nucleolus is practically impossible due to its complexity, suggesting that the least core (approximation to the core) is a more computationally accessible method for analyzing feature importance or value allocation in cooperative settings. This work also suggests there may be limitations in generalizing these findings across different types of data or models, given the specific nature of the computational improvements and the contexts in which they were tested. In this dissertation, I extend the Core and Nucleolus feature importance methods by applying these methodologies to linear models (considering Seatpos and Predator-prey datasets). I have also employed logistic regression models feature importance assessment by considering the entire dataset. The following sections describe these methods.

Shapley feature importance (SFI)

This section presents the Shapley value feature importance method (SFI).

Shapley value has been a popular solution concept for interpreting ML models. Shapley net effects developed by Lipovetsky [37] is a pioneer work that estimates the marginal contribution of the features considering the linear regression model. SHAP (SHapley Additive exPlanations) by Lundberg and Lee [15] is a method to explain individual predictions. Faith-Shap extends the concept of Shapley values specifically for attributing importance to both individual features and their interactions within black-box models [214]. This approach uses higher-order polynomial approximations to faithfully represent the value function of a model, leading to a unique and computationally efficient way to understand and explain the contributions of both individual features and their interactions to the model's predictions. IME (Interactions-based Method for Explanation) provides explanations for individual predictions of classification models [193]. Causal QII [211] measure accounts for correlated inputs or joint influences while measuring the feature influence. Shapley net effects [37] is the base model used in the scope of this dissertation. Shapley net effects consider R-squared values from a linear regression model with multicollinearity issues to measure the feature importance values. The following Shapley feature importance algorithm (Algorithm 1) is derived from Shapley net effects, which was originally applied within the framework of linear regression models. SFI extends the application of the Shapley value to the logistic regression model. SFI serves as a basis to compare the remaining feature-importance methods developed in the scope of this dissertation.

The SFI algorithm starts by determining different combinations of features and running models for each combination of the features. The run ends by returning the model summaries and extracting model performance values, such as $\mathsf{R}_{\mathsf{squared}}$ values. The newly obtained $\mathsf{R}_{\mathsf{squared}}$ values are used as the new data to define the game in characteristic form. For example $\nu(\{1\})$ is assigned the $\mathsf{R}_{\mathsf{squared}}$ value of feature 1, $\nu\{12\} = \mathsf{R}_{\mathsf{squared}}$ of feature 1 and feature 2 and so forth. The next step of the algorithm defines the game for n features

```
Algorithm 1: Shapley value feature importance values (SFI)
Data: $D(X_1, X_2, ..., X_n)$
$R_{squared} \leftarrow []$;
combinations $C \leftarrow []$;
for $X_i$ in range $(1 : n)$ do
    determine combinations of features;
    for a combination $C$ in the list of combinations do
        run regression models;
        get model summaries to extract $r_{squared}$ values
    end
end
Data: $(v\{1\}, v\{2\}, ..., v\{allfeatures\})$
define the game $(n, V)$ in characteristic form;
calculate Shapley regression values using Eq [4];
end when all permutations of coalitions are evaluated;
Return Shapley feature importance values.
```

given the respective characteristic values, i.e., coalition values v in lexicographic order. This is the second set of data used to compute the Shapley feature importance values. The algorithm ends when all permutations are evaluated, and the data about features' marginal contributions is obtained.

Core feature importance - CorFI

In this section, the concept of stability and the link between the core and the feature importance measure is explored. Kalousis et al. [215] describe the stability of feature as a quantification of how different training sets affect feature preferences, which take the form of a subset of selected features or alternatively of a weighting-scoring or a ranking of the features. In other words, the stability helps to determine how sensitive the model outcome will be to variations in the training set. A major challenge in ML is that no single agreed measure is used to quantify stability [216]. Many important features could be selected for the model. How stable would this selection be? For example, in a regression model, including a large number of features will artificially increase the R^2 value of the model. Does this increase in the R^2 value mean the model with all features, i.e., the grand coalition, should form, and this is the stable solution, or would some subset models be more accurate and reliable for the prediction?

To address these questions, the core concept could come in handy. Value stability in ML implies a model or an outcome with a robust prediction of stochastic changes. This refers to if varying the feature subset varies the model recommendations and suggestions. The stability concept is vital to answer if and how much we can trust the model outcomes. If including a new feature in the model drastically changes the prediction, perhaps we will not be able to trust the initial output of the model as a true representation of the problem.

Prior work suggests several approaches to measure the stability of feature selections [215, 217, 21]. The goal of these approaches is to enhance the statistical feature selection methods with concurrent analysis on stability to improve the quality of the selected features [218].

The core of cooperative game theory could be a way to analyze the stability of the feature importance values and select the most reliable set of features for the prediction. Yan and Procaccia [21] consider core for a credit assignment problem (feature and data valuation) and show that the least core - relaxation of the core that is always feasible, can be useful and sometimes preferable alternative feature importance measure over Shapley values. Unfortunately, it is possible that the core does not exist, meaning there are some models in which there aren't any stable payment allocation profiles that can be allocated to the features. This could be why the core has not been widely considered to determine the stability of feature importance. The possibility of an empty core could mean ”competition” and a conflict of interest between the features, indicating that the features have a high substitution rate. Therefore, forming a stable coalition between highly interchangeable features would be challenging. Yan and Procaccia [21] provide a way to tackle this limitation by using the least core.

The algorithm below presents the steps to compute the core:

The necessary conditions for the core are:

Algorithm 2: Core feature importance (CorFI) values

```
Data: $D(X_1, X_2, ..., X_n)$
A characteristic function $\nu : 2^N - > R$, where $N$ is the set of features.;
Initialize the set of features P = N and the core $C = \{\}$;
while P is not empty do
    Remove a player $i$ from $P$ ;
    Compute the worth without $i$, $W_i = \sum_{S \subseteq N\ i} \nu(S)$;
    If $W_i \geq V(N)$, add $i$ to $C$ ;
    Otherwise, continue with the next feature in P ;
    Update P to be the set of the remaining features after removing all coalitions
      that contain i ;
    Update P to be the intersection of P and the remaining feature in $N$ after
      removing all coalitions that contain $i$ ;
end
Return Return $C$ as the core feature importance values.
```

1. Efficiency: The total reward is fully distributed with no waste.

2. Pareto Optimality: There is no other allocation that could make at least one player better off without making at least one other player worse off.

3. Stability (No blocking coalitions): There is no subset of players (coalition) that can break away and secure themselves a better outcome by redistributing the reward among themselves according to their own coalition's value, leaving the rest of the players with less than they would receive in the proposed allocation.

Emptiness of the CorFI

The core emptiness in feature importance may stem from a lack of diversity or importance in the considered features. Some features may be redundant, highly substitutable, or insufficiently informative to contribute to the model's performance, making it challenging to identify a stable set of features. The following points offer insights into why machine learning models might encounter situations with an empty core:

Lack of synergistic effect: The empty core can indicate that the features do not have a synergistic effect on the model's performance. In other words, no combination of features leads to a better outcome for all players. This may occur if the features do not provide complementary information or utility. When the core is empty, it implies that there is no set of features that is both efficient (i.e., each feature contributes to the outcome) and fair (i.e., each feature receives a fair share of the outcome).

Non-cooperative setting: In some settings, the core may be empty because features act independently, without necessarily collaborating to achieve a common goal. This could occur when each feature represents a different aspect of the problem and does not depend on or interact with other features. Cooperative game theory assumes that features can form coalitions and work together to achieve a common goal, but this may not hold in a non-cooperative setting. In this case, the empty core may reflect the absence of a stable solution that satisfies fairness properties. For instance, in a dataset where each feature captures a different characteristic of a customer's purchase history, such as the amount spent, the number of purchases made, and the time of purchase, each feature may be important independently of the others.

Noise or variability in data: If there is a significant amount of noise or variability in the data, it may be difficult to identify a stable set of features that lead to a fair allocation of payoffs. In this case, the empty core may suggest that the data is too noisy or variable to allow for a stable solution.

Biased or incomplete data: If the data used to train the machine learning model is biased or incomplete, it may be impossible to identify a stable set of features that lead to a fair allocation of payoffs. In this case, the empty core may suggest that the data used to train the model is not representative enough to allow for a stable solution. For example, in a healthcare setting, suppose that a hospital wants to allocate medical resources fairly among different patient groups. However, certain patient groups may be underrepresented in the available data, leading to a lack of information about their needs and characteristics. In this case, the resulting resource allocation model may not be able to allocate resources fairly to all patient groups, and the core may be empty. If the data mainly includes information about the needs of younger patients, the resulting model may be biased towards allocating more resources to that group, even if older patients require more urgent medical attention. This highlights the importance of ensuring that the available data is representative of all patient groups to ensure a fair and stable allocation of medical resources.

Finally, it is also possible that the empty core is a consequence of the particular dataset and that a different data or machine learning algorithm would yield a non-empty core. Some approaches could be employed to turn the empty core into a non-blocking coalition, such as endogenously by forming new beliefs and attitudes or exogenously from an external intervention [219].

Nucleolus feature importance - NcFI

In this section, the Nucleolus feature importance (NcFI) method is presented.

The Nucleolus, as described in Section 2.3.2, is the set of efficient and individually rational vectors, that is, the gain that players in coalition S can obtain if they leave the grand coalition N under the imputation x and instead take the payoff $v(S)$ [111, 112]. The properties of the Nucleolus, based on min-max fairness and stability, could be suitable to determine models with the essential features. The min-max notions of fairness offer an alternative approach by "leveling-up," i.e., prioritizing improving the model's performance on the group for which performance is the worst [111]. Such optimizations reduce the performance of a model if it improves the performance of another model that is worse off [220]. A more detailed description of the Nucleolus can be found in Subsection 2.3.2.

The steps of finding the Nucleolus are to find a vector $x = (x_1, x_2, ..., x_n)$ that minimizes the maximum of the excesses $e(x, S)$ over all S subject to $x_j = v(N)$. The process of minimizing the maximum of a collection of linear functions subject to a linear constraint is converted to a linear programming problem [220]. Second and more linear programming problems may be used to minimize the next largest excess until n-tuple imputation x is found.

The Nucleolus could be a useful measure of feature importance by identifying which groups contribute the most to the overall loss or dissatisfaction in the game. It evaluates the degree of dissatisfaction or excess (also known as "loss") for each coalition and determines a fair distribution of the payoff based on these values. In other words, it seeks to minimize the maximum loss among all coalitions. This can be achieved by following the steps outlined in Algorithm 3, which involves computing the pro-rata imputations (proportional allocation based on the contributions), assigning joint worth to coalitions, calculating excess values, and iteratively adjusting the imputations until the excess is minimized. Pro rata allocations are computed as follows: $\mathsf{pro} - \mathsf{rata}_i = \frac{\nu(i)}{\sum_{i=1}^{n} \nu(i)} \nu(N), i \in N$. The nucleolus is attractive as it is unique and always exists.

Algorithm 3: The Nucleolus allocation of feature importance values (NcFI)

```
Data: $D(X_1, X_2, ..., X_n)$
Data: A characteristic function $\nu : 2^N - > R$, where $N$ is the set of features
for S be the set of all non-empty coalitions do
    compute the pro-rata imputation $x_S$ using the formula: $x_S = \nu(S)/|S|$;
end
for each feature $i$ do
    Assign their worth to all coalitions that include $i$ to obtain the joint worth for
      each coalition;
end
for coalition S in S do
    Calculate the excess e(x, S) using the formula: $e(x, S) = max\{\nu(T) - x(T)|T$ is
      a subset of $S\}$;
end
Initialize x to any feasible imputation ;
while there exists a feature i and a coalition S in S such that i is in S and
  $e(x, S) > \nu(\{i\} \subset S)/(|S| + 1)$ do
    Increase $x_i$ by $(\nu(i \cup S) - x(S))/(|S| + 1)$;
    For each coalition T in S that contains i, Recalculate the $x_T$ pro-rata
      imputation and the excess $e(x, T)$;
end
Return $x$ as the Nucleolus imputation vector.
```

For the feature importance, the Nucleolus evaluates which group has the highest unhappiness or excess, i.e., the loss, and based on this, determines the respective payoff distribution. What does unhappiness or excess translate in Nucleolus for feature importance problem? Think of a model where the value of the model $\nu(x_1)$ is 0.5, the value of another model with $\nu(x_2)$ is 0, while the value with these two features $\nu(x_1, x_2)$ is 0.9. Looking at this, the $\nu(x_2)$ has no importance for the prediction, while it adds some value when used with another feature $\nu(x_1)$. In this way, it helps the other feature to provide more explanation about the target. This could be due to some apparent, confounding, or latent associations between the features and the target; therefore, distributing the excess of the performance in a way that reflects the x_2's contribution is essential. For example, it is possible x_2 is significantly associated with the omitted feature x_1 and reflects the effect of the omitted feature in addition to its effect. When the omitted feature x_1 is included in the model, the originally non-significant feature no longer captures the partial effect of the omitted feature. However, it now reflects the ”true” effect of that feature, which is significantly associated with the target. According to the Nucleolus solution, the feature will be ”unhappy” if it gets an importance score of 0 by reflecting the prediction between itself and the target alone.

A criticism of Nucleolus and general min-max approaches is that they may excessively focus on improving the performance of a specific model j, sacrificing another model [221, 222]. This criticism is reasonable, and numerous approaches were proposed to deal with this limitation [223, 224, 225]. Improving this limitation is not within the scope of this research effort; instead, our goal is to estimate the nucleolus feature importance results obtained from a generic nucleolus solution.

3.4.2 VOTING GAMES BASED FEATURE IMPORTANCE

This section explores methods for assessing feature importance derived from voting game theory.

Voting games-based feature importance, a concept explored in Section 2.3.3, could be used to assess the importance of individual features within machine learning models. The objective of these games is to quantify feature importance values by leveraging the mechanics of voting systems where each feature is assigned a weight representing its influence, and there is a threshold value (model performance) that can be reached. This threshold signifies a specific level of model performance (such as R-squared values, accuracy, precision, recall, or any other relevant metric) that the combination of features must achieve or exceed for the outcome to be considered successful. This threshold may also serve as an indicator of the point at which the features fail to achieve the desired outcome. The following sections delve into two distinct types of voting games—the Shapley-Shubik and the Banzhaf power index—which were explored to measure the importance of features.

Shapley-Shubik Feature Importance - SH2FI

This section introduces the feature importance method based on the Shapley-Shubik power index (SH2FI).

As a feature importance method, the Shapley-Shubik index can be used to determine winning and losing models, thereby distinguishing between models that exhibit good or bad performance. Good performance refers to models that are effective, accurate, or successful in their predictions or classifications. In contrast, bad performance refers to models that are ineffective and inaccurate in predicting the relationship between the target and the interest variables. SH2FI can be used to evaluate the power of features given these model performances when these features are added following some sequence of order and threshold value (quota). The threshold value can be interpreted as the level of importance required for a feature to be considered relevant or significant. The order of features considered in a model plays an essential role in its contribution when there are overlaps among the features. This is the same logic as the hierarchical models, i.e., hierarchical regression.

Consider the following coalitions with features $\{x_1, x_2\}$ and $\{x_2, x_1\}$. These coalitions may be equivalent in some scenarios as they contain the same element. In other cases, when the players join sequentially, the order of joining may significantly affect the outcome. $< \mathsf{P2}, \mathsf{P1}, \mathsf{P3} >$ is an example of a sequential coalition, where the player joins the coalition in the following order: P2 joins the coalition first, P1 joins the second, and finally, P3 joins. <> notation is used instead of {} to distinguish sequential coalitions. This solution setup is known as the Shapley-Shubik index [122], which can be applied to general-purpose estimations, where the estimates are interpreted directly in terms of the a priori ability of the participants to affect the result. The a priori ability of the player defines the pivotal player that helps the sequential coalition to change from a losing coalition to a winning one. The losing or winning coalition is determined based on some threshold value or quota, which can be interpreted as the level of importance required for a feature to be considered relevant or significant.

To compute the SH2FI, One can proceed by following the steps outlined in the Algorithm 4.

Algorithm 4: Shapley-Shubik feature importance - SH2FI

```
Input: List of all sequential coalitions: $< \mathsf{P1}, \mathsf{P2}, ..., \mathsf{Pn} >$
Initialize pivotal_counts = {}, total_coalitions to 0;
for each coalition in $< \mathsf{P1}, \mathsf{P2}, ..., \mathsf{Pn} >$ do
    Determine the pivotal feature in the coalition;
    if pivotal feature not in pivotal_counts then
        pivotal_counts[pivotal feature] = 1;
    end
    else
        Increment the count for the pivotal feature in pivotal_counts;
    end
end
for each feature in pivotal_counts do
    Count how many times the feature is pivotal, and add this to total_coalitions;
end
for each feature in pivotal_counts do
    importance_value = count_of_pivotal / total_coalitions;
end
return SH2FI importance values for each feature;
```

SH2FI algorithm begins with a predefined list of all possible sequential coalitions, where each coalition represents a subset of features. The algorithm identifies the pivotal feature for each sequential coalition in the list. The pivotal feature is the one whose presence or absence affects the model's performance. SH2FI assesses how often each feature is identified as pivotal across all sequential coalitions. This involves counting the occurrences of each feature being pivotal. The counts obtained in the previous step are then converted into fractions by dividing each count by the total number of sequential coalitions. The final output measures each feature's importance based on its influence in different coalition settings.

Banzhaf power index feature importance (BFI)

Here, the Banzhaf power index is considered to measure feature importance values. Machine learning models could have high or low performances; with the Banzhaf power index, we begin listing all the coalitions of the models and then finding which coalitions are winning. Banzhaf power index has found substantial application in assessing feature importance, especially within the context of tree-based models [226, 227, 228, 229]. The work by Karczmarz [226] has demonstrated that the Banzhaf power-based feature importance method has a more intuitive interpretation, allows for more efficient algorithms, and is much more numerically robust. Banzhaf power index was also applied to explain feature importance methods for neural networks [230]. Banzhaf power index-based feature importance methods outperform current consistent random forests in terms of classification accuracy and are superior to, or at least on par with, Breiman's random forests, support vector machines (SVMs), and k-nearest neighbors (KNNs) classifiers [227]. Another study explores the application of voting games to random forest models and recommends extending this approach to linear regression models as a potential avenue for future research [231]. However, to the best of my knowledge, neither the Shapley-Shubik Power Index nor the Banzhaf Power Index has yet been applied to linear or logistic regression models, nor have been explored within the realm of agent-based models or in scenarios. The steps to compute the Banzhaf power index are presented in the Algorithm 5.

The BFI algorithm, as described in Algorithm 5, specifies the identification of all winning coalitions in a game, followed by counting how many times each feature is crucial to the formation of a successful coalition. The algorithm then converts these counts into

Algorithm 5: The Banzhaf-power index feature importance values (BFI)

```
Data: $D(X_1, X_2, ..., X_n)$
Construct models with single features using the data $D$;
Record model performances using the chosen metric;
List models with high performance (winning coalitions);
Initialize an empty dictionary D to store the BFI for each feature;
for Each feature i in the list of features P, do
    Initialize a counter variable count to 0;
    for each coalition c in the list of winning coalitions $C$ do
        If removing feature i from coalition c would cause it to become a losing
          coalition, increment count by 1;
        Compute the Banzhaf power index for feature i by dividing the count by
          the total number of winning coalitions ;
        Add the Banzhaf power index for feature $i$ to the dictionary D with the
          feature's name as the key ;
    end
end
Return the dictionary D with Banzhaf feature importance for each feature.
```

fractions or decimals, which represent the power of each feature in the game. In essence, the algorithm provides a way to measure the influence of each feature in the game by analyzing its contribution to the formation of winning coalitions.

Even though the features in the ML do not directly ”vote” for a candidate, the intuition and the objective of using the Banzhaf power index with ML models are to identify which coalitions (models) are capable of ”winning” by having a relatively better performance compared to the other models. Also, this measure helps determine the probability of the feature's ability to change the outcome with its influence. The features attempt to predict the target with combined powers, and when removing a feature from the model, shows the role of its direct vote, i.e., the influence on the model to be categorized as a winning one.

The Banzhaf power index feature importance could be particularly appropriate to measure the fairness of the distribution gains by determining the likelihood that (1) feature x_k is part of the winning coalition and that (2) x_k's contribution is necessary to achieve a certain level of performance. This certain level of performance is the quota or threshold necessary to determine if the feature yields a good (winning state) or bad (losing state) effect on the model's performance. This threshold τ can be chosen randomly.

3.4.3 CONFLICTING CLAIMS FEATURE IMPORTANCE

In this section, conflicting claims solution to measure feature importance values is discussed. Conflicting claims solutions, also called rules, are not as widely used, and to demonstrate their computations, examples are presented along with their algorithms.

Conflicting claims problems are used in situations that are described with claims, but there are not enough resources to distribute among all the claims due to resource constraints. These claims represent contributions towards the model's performance in a machine learning model. In such a scenario, a conflicting claim "rule" specifies how to divide the available resources, known as the endowment, which characterizes the model's performance. In the subsequent chapter 4 detailing the experiments, I tried different endowment values to observe their effects on the feature importance values.

In the following subsections, conflicting claims feature importance methods are discussed. These encompass Constrained Proportional feature Importance (CPI), Constrained EQual Awards feature importance (CEqA), Constrained EQual Losses feature importance (CEqL), Conflicting Claims Talmud Valuation (CCTV), Conflicting Claims Random Arrival (CCRA).

Constrained Proportional feature Importance (CPI)

Here, the Constrained Proportional feature Importance (CPI) method is discussed.

In the context of feature importance values, the Proportional (P) rule from conflicting claims could be used to determine how much weight each feature should be given in a predictive model. For instance, if there are three features with equal strength in predicting the outcome, the P rule would recommend that each feature be given equal importance. However, if one feature is judged twice as important as the others, the P rule would recommend that this feature be given twice the weight of the other features. In short, the P rule suggests that the importance or weight given to each feature should be proportional to its relative strength in predicting the outcome.

```
Algorithm 6: Conflicting claims proportional feature importance (CPI)
Data: $E$ - total endowment; $c_i$, claim of each feature $i$; $N$, the set of all features
      with claims
Construct models with single features using the data $D(X_1, X_2, ..., X_n)$;
Record model performances using the chosen metric as claims $c_i$;
Calculate the total sum of claims $\Sigma = \sum_{i \in N} c_i$;
Calculate the proportionality factor $\lambda = \frac{E}{\Sigma}$;
for each feature $i \in N$ do
    Calculate proportional distribution $P_i(E, c) = \lambda \cdot c_i$;
end
Return the set of CPI $P_i(E, c)$ for all features.
```

Algorithm 6 presents the steps to compute conflicting claims proportional feature importance values. The total sum of claims called Sigma Σ is calculated by summing up the claims of all features c_i within the set of features N. The proportionality factor λ is determined by dividing the total endowment E by the total sum of claims Σ. This factor ensures that the distribution is adjusted in proportion to the total endowment available. For each feature in the set N, the algorithm calculates the CPI - proportional feature importance values $P_i(E, c)$ by multiplying the claim of feature i (c_i) by the proportionality factor λ.

Example: Consider the following example. The initial claim of Feature 1 is 0.3. This claim represents the performance of the model given the feature's contribution used in the model independently without any other feature. Similarly, Feature 2 exhibits a model performance of 0.5, while Feature 3's performance claim reaches 0.8. For endowment $E = 1$ the claims are [0.3,0.5,0.8],

Step 1: Calculate total sum of claims

First, sum the claims of all features:

$$\Sigma = \sum_{i \in N} c_i = 0.3 + 0.5 + 0.8 = 1.6$$

Step 2: Calculate proportionality factor

Next, calculate the proportionality factor λ:

$$\lambda = \frac{E}{\Sigma} = \frac{1}{1.6} = 0.625$$

Step 3: Calculate proportional distribution for each feature

Now, for each feature, calculate the proportional distribution $P_i(E, c)$:

For $c_1 = 0.3$:

$$P_1(E, c) = \lambda \cdot c_1 = 0.625 \cdot 0.3 = 0.1875$$

For $c_2 = 0.5$:

$$P_2(E, c) = \lambda \cdot c_2 = 0.625 \cdot 0.5 = 0.3125$$

For $c_3 = 0.8$:

$$P_3(E, c) = \lambda \cdot c_3 = 0.625 \cdot 0.8 = 0.5$$

CPI values

The constrained equal proportional feature importance values given the endowment $E = 1$ among the features based on their claims are:

- $P_1(E, c) = 0.1875$ for the first feature,
- $P_2(E, c) = 0.3125$ for the second feature,
- $P_3(E, c) = 0.5$ for the third feature.

These results show how the total endowment of 1 is distributed among the features according to their claims, ensuring each feature receives a portion of the endowment proportional to its claim. This way, the feature with the highest claim ($c_3 = 0.8$) receives the largest share of the endowment (0.5), while the feature with the lowest claim ($c_1 = 0.3$) receives the smallest share (0.1875). The distribution reflects the relative magnitude of each feature's claim to the total available endowment.

Constrained EQual Awards feature importance (CEqA)

In this section, the Constrained EQual Awards feature importance (CEqA) method is presented.

The constrained equal awards (CEA) rule distributes the importance score equally among features, considering their claims based on their contribution to the model performance. The method is based on principles of equity and fairness, with constraints ensuring that the allocation does not exceed available resources. The overall ”resource” in the constrained equal awards can be conceptualized as the importance score that must be distributed across the various features. Each feature ”claims” a certain amount of importance based on its contribution to the overall model performance. The CEA rule then suggests that we distribute the importance score equally among all the features but not exceeding each feature’s claim. The main limitation of the CEA technique is that it assumes that all features have an equal claim to the target variable, which may not be the case in reality. Below, the steps for Constrained EQual Awards feature importance (CEqA) are presented.

Algorithm 7: Constrained Equal Awards feature importance (CEqA)

```
Data: $E$ - total endowment; $c_i$, claim of each feature $i$; $N$, the set of all features
      with claims
Construct models with single features using the data $D(X_1, X_2, ..., X_n)$;
Record model performances using the chosen metric as claims $c_i$;
Initialize $\mu$ with an estimated value, e.g., $\mu = E/|N|$;
while the sum of $\min\{c_i, \mu\}$ for all $i \in N$ does not equal $E$ do
    Adjust $\mu$ to ensure $\sum_{i \in N} \min\{c_i, \mu\} = E$;
end
for each claimant $i \in N$ do
    Calculate $CEA_i(E, c) = \min\{c_i, \mu\}$;
end
Return the set of $CEA_i$ for all claimants.
```

The total endowment, as outlined in Algorithm 7, is allocated randomly, serving as a rational representation of the model performance metric. For instance, if the R-squared value is selected as the performance metric for the model, then the endowment E would range between 0 and 1. This range would capture the extent of variance explained by the model. The performance of the model when utilizing only a single feature constitutes the claim of that feature.

Example: Consider the following example. The initial claim of Feature 1 is 0.3. This claim represents the performance of the model given the feature's contribution used in the model independently without any other feature. Similarly, Feature 2 exhibits a model performance of 0.5, while Feature 3's performance claim reaches 0.8. For endowment $E = 1$ and the claims [0.3,0.5,0.8], the following CEqA feature importance values will be allocated [0.3,0.35,0.35]. The first feature receives its full claim of 0.3 since it is less than the adjusted μ, which is approximately 0.333. The second and third features each receive an award of approximately 0.35, which is the adjusted μ value, ensuring the total awards do not exceed the endowment and respect each feature's maximum claim.

Constrained EQual Losses feature importance (CEqL)

Here constrained equal losses feature importance (CEqL) method is described.

CEqL allocates losses equally among features, identifying the set of features that contribute equally to the model performance or the error metric. In the context of feature importance, the CEL rule can be used to identify the set of features that contribute equally to a loss function or performance metric. These features can then be considered equally important in terms of their contribution to the error.

Algorithm 8: Constrained equal losses feature importance (CEqL)

```
Data: $E$ - total endowment; $c_i$ - claim of each feature $i$; $N$, the set of all features
      with claims
Build models with the initial data $D(X_1, X_2, ..., X_n)$ ;
Extract performance metrics ($c_i$) claim of each feature;
Initialize $r$ such that $\sum_{i=1}^{n} \max(0, c_i - r) = E$;
for $i = 1$ to $n$ do
    Allocate to feature $i$: $\max(0, c_i - r)$;
end
Return the set of $CEqL_i(E, c)$ for all features.
```

Algorithm 8 presents the steps to compute the constrained equal losses feature importance values, where E presents the estate or endowment, which is to be distributed among n features. For each feature labeled i, their respective claim is indicated by c_i. It is commonly the case that $\sum_{i=1}^{n} c_i > E$, signifying that the total of all claims exceeds the available estate, thereby rendering it inadequate to fulfill every claim fully.

Example: Consider the same example discussed for CPI and CEqA. The initial claim of Feature 1 is 0.3. This claim represents the performance of the model given the feature's contribution used in the model independently without any other feature. Similarly, Feature 2 exhibits a model performance of 0.5, while Feature 3's performance claim reaches 0.8. For endowment $E = 1$ and the claims [0.3,0.5,0.8], the CEqL feature importance values for this problem will be $[0.1, 0.2, 0.6]$. The computation of CEqL values is as follows: first, the total claim is determined $Sum = 0.3 + 0.5 + 0.8 = 1.6$. Next, this is deducted from the endowment E, $1.6-1 = 0.6$, which represents the loss. This loss is equally divided among the N (number of features) $0.6/3 = 0.2$. Finally, the average loss is subtracted from the initial claims to determine the final allocation for each feature, i.e., $(0.3 - 0.2 = 0.1), (0.5 - 0.2) = 0.3, (0.8 - 0.2 = 0.6)$.

Conflicting Claims Talmud Values (CCTV)

This section presents conflicting claims Talmud valuation (CCTV), which allocates the feature importance values by evaluating if the sum of all claimed feature importance values exceeds the total importance score available, then, the CEqA computation is applied to allocate the importance score proportionally to the claims of each feature. On the other hand, if the endowment is sufficient to satisfy the half-sum of the claims, then each feature is assigned an importance value equal to half of its claimed importance value. The remaining importance score is then allocated among the features using the constrained equal losses (CEqL) feature importance computation, which ensures that all features incur equal losses subject to no feature receiving a negative amount. Algorithm 9 presents the steps to computate CCTV.

Algorithm 9: Conflicting claims Talmud valuation (CCTV)

```
Data: $E$ - Endowment; $c_i$ - claim of each feature $i$; $N$, the set of features with
      claims
Calculate the half-sum of claims $H = \sum_{i \in N} c_i/2$;
if $E \leq H$ then
    Apply the Constrained Equal Awards (CEqA) rule;
    for each $i \in N$ do
        $T_i(E, c) \leftarrow CEqA_i(E, c/2)$
    end
    else
        Each player receives half of her claim;
        for each $i \in N$ do
            $T_i(E, c) \leftarrow c_i/2$
        end
    end
    Apply the CEqL for the remaining endowment;
    for each $i \in N$ do
        $T_i(E, c) \leftarrow c_i/2 + CEqL_i(E - H, c/2)$
    end
end
```

Example: Consider the same example discussed for CPI and CEqA. The initial claim of Feature 1 is 0.3. This claim represents the performance of the model given the feature's contribution used in the model independently without any other feature. Similarly, Feature 2 exhibits a model performance of 0.5, while Feature 3's performance claim reaches 0.8. For endowment, $E = 1$ and the claims [0.3,0.5,0.8], the CCTV allocation initially amounts to 0.15, 0.25, and 0.4, respectively, because the half sum of the claims of 0.8 is less than the total endowment of 1. Consequently, a surplus of 0.2 remains. This remaining endowment is then distributed in accordance with the CEqL discussed above, and the final allocation will be $[0.15, 0.35, 0.5]$.

Conflicting Claims Random Arrival (CCRA)

This section presents conflicting claims random arrival feature importance (CCRA) method. The concept of random arrival (RA) is similar to the Shapley value, Shapley-

Shubik, and Banzhaf power index, where the order of adding or removing features from the model can be crucial. Like these methods, the random arrival considers the order in which the features are added or removed from the model and how their contributions affect the model's performance. However, the RA rule differs in that it accounts for the order of arrival of the features rather than their coalition values or marginal contributions. Also, this method distinguishes itself primarily through the inclusion of an endowment.

Algorithm 10: Conflicting claims random arrival (CCRA)

```
Data: $E$ - Endowment; $c_i$ - claim of feature $i$; $N$ - set of features
Result: RA allocation vector where $RA_i$ is the allocation for feature $i$
Initialize an empty vector $V$ to store the sum of awards for each feature;
Initialize $V_i = 0$ for each $i \in N$;
foreach permutation $\pi$ of $N$ do
    $E_{temp} \leftarrow E$ ;                // Temporary endowment for each permutation
    foreach feature $i$ in $\pi$ do
        $award \leftarrow \min\{c_i, E_{temp}\}$;
        $V_i \leftarrow V_i + award$;
        $E_{temp} \leftarrow E_{temp} - award$;
    end
end
foreach claimant $i$ in $N$ do
    $RA_i \leftarrow \frac{V_i}{|N|!}$ ;                // Average award for each feature
end
Return the set of $CCRA_i$ for all features.
```

Consider an example where the endowment, $E = 1$, and the claims of the features are [0.3, 0.5, 0.8]. To determine the conflicting claims random arrival feature importance (CCRA) values first, the permutations and their corresponding allocations are detailed as follows:

Permutation: 0.3, 0.5, 0.8

- 0.3 receives: 0.3 (full claim, endowment remaining = 0.7)
- 0.5 receives: 0.5 (full claim, endowment remaining = 0.2)
- 0.8 receives: 0.2 (partial claim, endowment depleted)

Permutation: 0.3, 0.8, 0.5

- 0.3 receives: 0.3 (full claim, endowment remaining = 0.7)
- 0.8 receives: 0.7 (partial claim, endowment depleted)
- 0.5 receives: 0 (no endowment remaining)

Permutation: 0.5, 0.3, 0.8

- 0.5 receives: 0.5 (full claim, endowment remaining = 0.5)
- 0.3 receives: 0.3 (full claim, endowment remaining = 0.2)
- 0.8 receives: 0.2 (partial claim, endowment depleted)

Permutation: 0.5, 0.8, 0.3

- 0.5 receives: 0.5 (full claim, endowment remaining = 0.5)
- 0.8 receives: 0.5 (partial claim, endowment depleted)
- 0.3 receives: 0 (no endowment remaining)

Permutation: 0.8, 0.3, 0.5

- 0.8 receives: 0.8 (full claim, endowment remaining = 0.2)
- 0.3 receives: 0.2 (partial claim, endowment depleted)
- 0.5 receives: 0 (no endowment remaining)

Permutation: 0.8, 0.5, 0.3

- 0.8 receives: 0.8 (full claim, endowment remaining = 0.2)
- 0.5 receives: 0.2 (partial claim, endowment depleted)
- 0.3 receives: 0 (no endowment remaining)

Given the permutations and the allocation for the feature with a claim of 0.3 across all permutations, we have:

1. Permutation 0.3, 0.5, 0.8: Allocation is 0.3

2. Permutation 0.3, 0.8, 0.5: Allocation is 0.3

3. Permutation 0.5, 0.3, 0.8: Allocation is 0.3

4. Permutation 0.5, 0.8, 0.3: Allocation is 0

5. Permutation 0.8, 0.3, 0.5: Allocation is 0.2

6. Permutation 0.8, 0.5, 0.3: Allocation is 0

The average allocation for the claimant with a claim of 0.3 is computed as follows:

$$\text{Average Allocation} = \frac{\sum \text{Allocations}}{\text{Number of Permutations}} = \frac{0.3 + 0.3 + 0.3 + 0 + 0.2 + 0}{6} = 0.183\overline{3}$$

Thus, the average allocation for the feature with a claim of 0.3, computed step by step across all permutations, is approximately 0.183.

Similarly, for the feature with a claim of 0.5, the average allocation is approximately 0.283. For the feature with a claim of 0.8, the average allocation is approximately 0.533.

3.4.4 FEATURE IMPORTANCE FOR AGENT-BASED MODELING

This section presents the method to address one of the research questions introduced earlier (Section 1.3), focusing on the use of cooperative game theory-based methods for feature importance in analyzing empirical data collected about the simuland (system under study) as an input for agent-based modeling. This work described in this section and the respective results have been published in Winter Simulation Conference [177]. A detailed description of this approach is presented below.

Evaluating feature importance in empirical data for agent-based models is crucial for refining the development of simulation models. This process identifies key features to prioritize during the simulation design phase, guiding developers on where to focus their efforts for maximum impact and efficiency.

Feature importance evaluation of empirical data for the agent-based model could be useful knowledge in the development of the simulation model, i.e., which features they should focus their attention on during the simulation design, as well as the sensitivity analysis stage of a simulation project. This is important because developing targeted sensitivity analysis methods has been identified as a key challenge for agent-based modeling [40]. Moreover, the feature importance method can also assist in identifying and eliminating irrelevant features, which could lead to model complexity and reduced simulation performance. This method employs a feature importance evaluation algorithm to calculate the importance values of features from the behavioral space, resulting in a list of the most significant features that impact the system behavior. By focusing on important features, simulation designers can create more efficient models and avoid wasting resources on simulating irrelevant features, which could result in longer simulation run times and higher computational costs [177]. Additionally, the knowledge gained from identifying important features can also facilitate the parameterization of the simulation model, which is crucial for model calibration and validation. With this information, simulation designers can more accurately set and test the parameters of the model, leading to improved confidence in the model's results and predictions. Overall, this approach can help streamline the simulation design process and lead to more explainable and accurate simulation models.

Algorithm 11 presents the sequential steps for agent-based model feature importance values. Initially, it acquires data from either simulation model outputs or empirical datasets. Following this, the algorithm applies specific methods to compute the feature importance

Algorithm 11: ABM: Shapley value feature importance explanations

```
1 Input: Collect data about the simuland (in this study, Predator-prey dataset was
    used collected by Blasius et al., [176]);
2 Apply and compute feature importance methods (SFI, NcFI, SH2FI, etc.);
3 Return: Feature importance values;
4 Design or adjust the ABM simulation model based on the feature importance
    observations;
  Output: Optimized ABM simulation model.
```

values, ultimately generating and returning these calculated values for further analysis and application. This process facilitates the understanding of the significance of different features within the model. The data and the R code can be accessed `https://github.com/grigoryangayane/Predator-prey-model-feature-importance`.

TABLE 4: Single observation instances extracted from datasets

Exp/features	rotifers	algae	egg-ratio	eggs	dead animals	external
Exp1	5.42	0.83	0	0	0.4	NA
Exp5	9.83	0.73	0.27	2.61	0.4	NA
Exp8	11.04	2.83	0.67	7.42	0.4	160
Exp10	6.82	2.03	0.06	0.4	0.2	NA

Simulation models and various applications, such as cybersecurity [232], healthcare [233], and coalition formation [172, 234] that can generate a feature space are suitable for analysis. Table 4 presents single-sample observations captured from distinct experiments.

Within the framework of this dissertation, the initial three steps have been executed, and the results are presented in Section 4.4. The methods considered are Shapley feature importance, Nucleolus feature importance, Shapley-Shubik feature importance, Banzhaf feature importance, and conflicting claims future importance methods.

3.5 SHANNON ENTROPY-BASED PERMUTATION RELATIVE IMPORTANCE EVALUATION (PRIME)

In this section, the Permutation Relative IMportance Evaluation (PRIME) metric is explored, another contribution of this dissertation. PRIME combines two established methods: permutation tests [82, 196] and weighted Shannon entropy [197, 198, 199, 200, 201]. This hybrid metric, combining permutation tests and weighted Shannon Entropy, assesses feature importance by analyzing how features influence outcomes when their values are shuffled. PRIME metric measures the consistency and uncertainty of feature importance rankings (when feature importance values are arranged in descending order) across different data permutations. Essentially, PRIME uses permutation testing to evaluate the impact of features under random rearrangements and applies weighted Shannon Entropy to measure the consistency of a feature's importance rankings. Also, PRIME aims to establish a metric that enables the comparison of various feature importance methods.

This need arises from the observation that the results of other existing approaches, such as Permutation Importance (PIMP) [82], a prevalent metric for evaluating feature importance methods, have not been consistently effective. With Permutation Importance, I have observed that a single feature's statistical significance can vary across different methods, leading to interpretative challenges. For instance, applying the Permutation Importance evaluation analysis to the experiments within this dissertation has revealed that while one feature importance method deems feature X to be statistically significant, another feature importance method regards it as insignificant.

Note that human-subject experiments were conducted to assess the effectiveness of the explainable artificial intelligence methods [31, 32]. However, the complexity and fluctuations in human cognition and interpretation posed challenges in quantifying the direct impact of these methods on user trust and comprehension. Adopting a quantitative approach for assessing the effectiveness of feature importance methods in XAI offers multiple benefits over exclusively depending on human-subject experiments. Notably, it facilitates the generation of measurable, objective, and reproducible findings, providing a solid foundation for evaluating and comparing the impact of XAI strategies [235, 236]. Consequently, these advantages underscore the importance and need for the development of the PRIME metric.

I have tested PRIME on the feature importance methods developed during this dissertation, examining datasets characterized by independent features and features with a high degree of multicollinearity. The results are presented in Chapter 4. The findings demonstrate that PRIME can effectively quantify the permutation-based uncertainties inherent in feature importance assessments. The subsequent sections delve into the methodologies and applications of permutation tests, Shannon entropy, weighted Shannon entropy, and the combinations of these methods.

Generally, the prior studies on assessing feature importance methods can be broadly categorized into four groups: 1) evaluating the (in)sensitivity of explanations to changes in the model or input [237, 238], 2) deducing the accuracy of explanations based on the decrease in model performance after removing features [196, 239], 3) assessing explanations within a controlled environment where the importance of features is partially known [240, 241, 13], and 4) analyzing explanations through the lens of human interpretation.

Weighted Shannon entropy permutation importance evaluation metric aligns with the first categories, leveraging known data on feature importance to determine the permutations under which feature importance scores remain stable or vary in response to modifications in inputs when data is randomly shuffled (permuted). Unlike the fourth category, my focus is solely on the precision of feature importance evaluation after the permutations without considering the comprehensibility of explanations to humans. Weighted Shannon entropy-based PRIME can be regarded as a way to verify the reliability of feature importance evaluations, acting as a preliminary step before engaging in more resource-intensive studies involving human participants.

3.5.1 PERMUTATION TESTS

This section explores the concept of permutation tests.

Permutations involve altering the values of a feature in the dataset and observing the resultant change in the model's performance. The permutation tests aim to assess how robust the original feature importance rankings are to changes in the data. These tests are used to highlight the impact of individual features on the model's predictions.

To assess the significance of features in machine learning models different permutation-based methods have been proposed [82, 196, 242, 243, 244, 245]. The Permutation Importance Method (PIMP) a widely used technique that was developed to address biases linked to permutation in assessing feature importance values [82]. PIMP employs multiple permutations of the outcome vector to determine the distribution of importance scores for each variable under conditions that are not informative. The observed importance's P-value is then used to improve the feature importance evaluation by constructing models that incorporate features deemed statistically significant. Kaneko [196] has introduced a cross-validation-based approach to permutation feature importance (PFI), which involves several steps. Initially, a model is constructed using training data. Subsequently, the algorithm calculates the reference score (rs) of the model on a designated validation dataset (VD), where the score could represent accuracy for a classifier or the determination coefficient (r^2) for a regressor. The essence of the PFI process lies in permutation iterations. For each feature i in the dataset and each repetition j, the algorithm randomly shuffles the values in the i–th column to create a corrupted version of VD ($\mathsf{CVD}_{\mathsf{i},\mathsf{j}}$). The model's score ($\mathsf{s}_{\mathsf{i},\mathsf{j}}$) on this corrupted dataset is then computed. The Permutation Feature Importance ($\mathsf{PFI}_{\mathsf{i}}$) for each feature is subsequently determined based on the change in performance caused by these permutations. PFI quantifies the impact of permuting individual features on the model's overall performance. Additionally, permutation tests have been applied to evaluate the confidence intervals and variance estimation of importance values [245, 246].

One of the challenges with permutation tests is their potential failure to capture the uncertainty in feature importance rankings [82]. When features are permuted, and the model's performance is evaluated, the results have inherent variability due to randomness. This variability can lead to uncertainty in determining the true importance of each feature. However, many implementations of permutation tests might not account for this uncertainty adequately. Another issue is that the rankings of feature importance derived from permutation tests may significantly vary or closely resemble the initial, unpermuted rankings, as observed from the experiments conducted in the scope of this dissertation. This can happen if the model is overly dependent on certain features or if there are correlations among the features. To the best of my knowledge, the issues pertaining to the uncertainty and consistency of feature importance rankings resulting from randomized permutations have not been addressed yet.

3.5.2 SHANNON ENTROPY

This section explores Shannon entropy, which is a concept from information theory introduced by Claude Shannon. Shannon entropy measures the average amount of uncertainty or disorder associated with a random variable [247]. Shannon entropy is measured as follows:

$$H(X) = -\sum_{i=1}^{n} P(x_i) \cdot \log_2(P(x_i)) \tag{14}$$

In this equation, $P(x_i)$ represents the probability of occurrence of the specific event x_i, and n denotes the total number of events. The sum extends over all possible events i from 1 to n, where $\log_b$ indicates the logarithm to the base b. This base is typically set to 2 for entropy measured in bits but can also be e for natural units (nats) or 10 for hartleys. Shannon entropy is described with additivity property [248, 249] which asserts that for two independent systems or random variables X and Y, with entropies $H(X)$ and $H(Y)$, the entropy of the combined system $X+Y$ is the sum of their individual entropies, expressed as $H(X+Y) = H(X) + H(Y)$.

Proof

$$
\begin{aligned}
H(X) &= -\sum_{x} p(x)\log p(x) \\
H(Y) &= -\sum_{y} p(y)\log p(y) \\
p(x,y) &= p(x)p(y) \quad \text{(independent)} \\
H(X,Y) &= -\sum_{x,y} p(x,y)\log p(x,y) \\
&= -\sum_{x,y} p(x)p(y)\log[p(x)p(y)] \\
&= -\sum_{x,y} p(x)p(y)(\log p(x) + \log p(y)) \\
&= \left(-\sum_{x} p(x)\log p(x)\right)\left(\sum_{y} p(y)\right) \\
&\quad + \left(-\sum_{y} p(y)\log p(y)\right)\left(\sum_{x} p(x)\right) \\
&= H(X) + H(Y)
\end{aligned}
$$

3.5.3 WEIGHTED SHANNON ENTROPY

This section explores weighted Shannon entropy, an extension of the classic Shannon entropy.

Weighted Shannon Entropy, an advanced concept derived from information theory, extends the foundational principle of Shannon entropy by incorporating weights for each possible outcome [197, 198]. This adaptation allows it to more accurately quantify the uncertainty inherent in the value of a random variable or the result of a random process. By adjusting for the varying significance or relevance of each outcome, Weighted Shannon Entropy offers enhanced utility in situations where not all outcomes are equally important or occur in equal frequency, thereby providing a more nuanced understanding of uncertainty in diverse scenarios [198, 199, 200, 201].

Weighted Shannon entropy has been used in various fields, including medical imaging [197, 198, 199, 200, 201]. For example, weighted Shannon entropy was proposed to enhance the detection of scatterer concentrations in tissues using ultrasound [198]. This approach involves assigning weights to different parts of the signal based on their contribution to the overall information content. By applying a weighted Shannon entropy approach, the authors have demonstrated improved sensitivity in characterizing scatterer variations, offering a more effective means of analyzing tissue microstructures compared to conventional entropy methods [198].

Another work [200] studies ultrasound entropy imaging for detecting and monitoring thermal lesions, different forms of Shannon entropy, including typical Shannon entropy (TSE), weighted Shannon entropy (WSE), and horizontally normalized Shannon entropy (hNSE), were explored. The WSE estimation utilizes the same probability distributions as TSE but applies w as the weight to enhance its sensitivity to changes in disorder. The estimation of weighted entropy is defined as follows:

$$\mathsf{H}W = -\sum_{y=y_{min}}^{y_{max}} w(y) \cdot \log_2[w(y)] \tag{15}$$

Where the parameters are interpreted similarly as in equation 14. $\mathsf{H}W$ corresponds to weighted Shannon entropy, and $w(y)$ is the weighted probability distribution of the specific events. The sum extends over all possible events i from 1 to n, where $\log_b$ indicates the logarithm to the base 2.

This study [200] found that WSE, which incorporates signal amplitudes as weights, offered improvements over TSE by achieving better detection performance in some areas. THE hNSE provided superior lesion detection accuracy and contrast, indicating the usefulness of frequency and normalization adjustments in weighted entropy applications for medical imaging.

Another study [201] has employed the Entropy Weight Method (EWM) to integrate different prediction models by assigning weights based on the dispersion of prediction errors. The methodology determines the importance of each model in improving prediction accuracy by evaluating the entropy value of model prediction errors. This approach has been applied to traffic flow prediction, illustrating its utility in combining models to achieve more accurate forecasts.

In various studies, weighted Shannon entropy calculations are refined through the application of weights, which are determined by several factors. These factors include the relevance of specific data points [199], the magnitude of prediction errors [201], and at times, subjective criteria reflecting expert judgment [198, 197]. For example, Qu et al., [201] suggest assigning weights to different models in a way that the smaller the entropy value of the prediction error of an individual model, the greater the weight should be assigned. Assigning different weights allows for a tailored analysis that prioritizes certain aspects of the data. These weights are crucial for emphasizing certain elements over others, thereby adjusting the entropy calculation to reflect the importance of specific aspects of the data being analyzed. Overall, the review of the prior work has shown that the exact method of determining weights can vary significantly between applications [199, 201, 198, 197].

After determining weighted Shannon entropy HW, the average weighted Shannon entropy $\langle HW \rangle$ [250] can be computed using Equation 11. This computation is instrumental in evaluating and comparing datasets and methods in terms of their average uncertainty and ability to produce consistent results.

$$\langle HW \rangle = \frac{1}{N} \sum_{j=1}^{N} HW_j \tag{16}$$

Where N represents the number of events or features being assessed for their significance in prediction. It is assumed that the probability distributions correspond to an equal number of features, and these distributions measure the same type of information. A higher entropy signifies increased uncertainty, randomness, or inconsistency in the feature importance ranking.

Weighted Shannon entropy has been previously discussed with permutations [251]. However, permutation in the discussed context refers to analyzing time series data by comparing neighboring values of each point (subsets of time series) to determine their ordinal patterns. This method, known as Permutation Entropy (PE), evaluates the complexity of signals by mapping them to sequences that reflect their underlying order. This approach is notable for its ability to remain robust against artifacts that occur at low frequencies, making it versatile for analyzing a wide range of time series data. While PE [251] was effective in identifying patterns within the time-series data, offering insights into its complexity without being heavily influenced by noise or linear distortions, it is important to note that PE is not related and does not measure feature importance values.

3.5.4 WEIGHTED SHANNON ENTROPY BASED PERMUTATION IMPORTANCE EVALUATION (PRIME)

This section describes the Permutation Relative IMportance Evaluation (PRIME) metric, which combines the concepts of permutation tests and weighted Shannon entropy, previously detailed in Sections 3.5.1 and 3.5.3.

I have followed the subsequent steps described in Figure 9 to formulate the permutation relative importance evaluation with weighted Shannon entropies (PRIME).

Figure 9 provides a comprehensive overview of the weighted Shannon entropy-based PRIME metric, detailing its essential elements.

1. *Actual data* refers to the initial or original data collected directly from real-world sources or derived from simulation models. Section 3.2 presents the data used in this dissertation.

2. *Permuted data* refers to the process of randomly shuffling the values within each feature across the dataset. This data refers to the permutations applied to the original

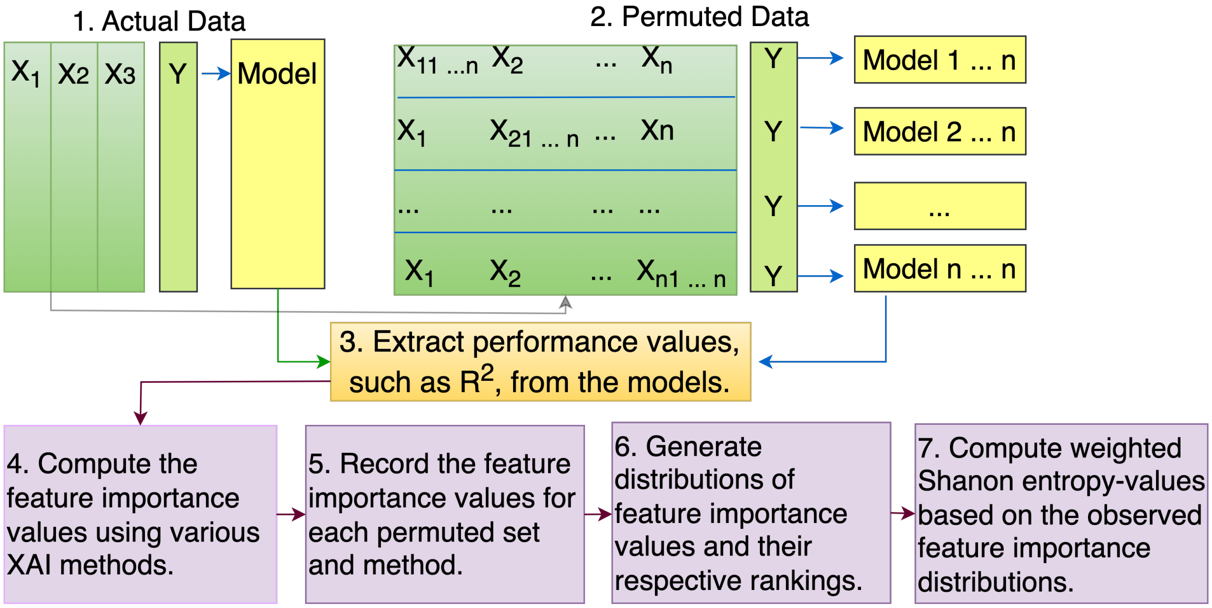
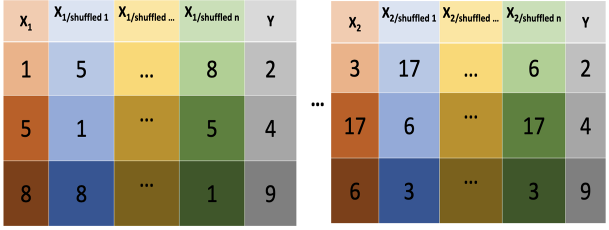

1. Actual Data
X_1
X2
X3
Y
Model
2. Permuted Data
$X_{11\ldots n}$ X_2 ... X_n
X_1 $X_{21\ldots n}$ ... Xn
... ...
X_1 X_2 ... $X_{n1\ldots n}$
Y
Y
Y
Y
Model 1 ... n
Model 2 ... n
...
Model n ... n
3. Extract performance values, such as R^2, from the models.
4. Compute the feature importance values using various XAI methods.
5. Record the feature importance values for each permuted set and method.
6. Generate distributions of feature importance values and their respective rankings.
7. Compute weighted Shanon entropy-values based on the observed feature importance distributions.

FIG. 9: Weighted Shanon entropy-based PRIME overview

(raw) dataset outlined in Section 3.2. The objective of this step is to break any true relationship between the feature and the target variable. This step is similar to the PFI approach [196]. Figure 10 presents the data permutation example.



X_1	$X_{1/shuffled\ 1}$	$X_{1/shuffled\ \ldots}$	$X_{1/shuffled\ n}$	Y
1	5	...	8	2
5	1	...	5	4
8	8	...	1	9

...

X_2	$X_{2/shuffled\ 1}$	$X_{2/shuffled\ \ldots}$	$X_{2/shuffled\ n}$	Y
3	17	...	6	2
17	6	...	17	4
6	3	...	3	9

FIG. 10: Data permutation example

3. *Construct models and extract their performance values* using the actual and permuted data. As outlined in Section 3.3, the models used in the scope of this dissertation are linear and logistic regression models.

4. *Compute the feature importance values* for all permuted features using the feature importance methods outlined in Section 3.4. For the Seatpos dataset, comprising 8 features, 30 permutations were evaluated, resulting in the evaluation of 240 models and the derivation of 240 feature importance scores. Each set of 30 scores corresponds

to the permutations of a single feature. For the Adult dataset, which comprises 9 features, 60 permutations were conducted. This process resulted in the assessment of 540 models and the generation of 540 importance scores. Each batch of 60 scores is associated with the permutations of a single feature. Finally, the Predator-prey dataset, containing 4 features, underwent 60 permutations, leading to the evaluation of 240 models. Correspondingly, 240 feature importance scores were generated. Each batch of 60 scores is associated with the permutations of a single feature.

5. *Record the feature importance values* for each permuted dataset and corresponding feature importance method. This data is subsequently used to create distributions of feature importance values.

6. *Generate a distribution of importance values.* In this step of the PRIME evaluation, the distribution of feature importance scores for each feature is generated based on each permutation by counting how many times each outcome occurs. The outcome here refers to the feature importance ranking that is obtained by ordering the feature importance values in descending order. These distributions present the overall variability of how each feature contributes to the model's predictions when it undergoes permutations.

7. *Computed weighted Shannon entropy* described in Algorithm 12 to measure the consistency of the feature importance ranking generated by various methods. This also facilitates the comparison of how well different feature importance methods can rank features as they undergo permutations. It underscores the methods' capacity to maintain accurate feature prioritization amidst the variability introduced by permutations.

Algorithm 12 comprises the following steps: Firstly, data collected from the permutations is used to calculate weighted probabilities by multiplying each rank's probability by its frequency. Feature importance ranks are obtained by ordering the feature importance values in descending order. Then, these values are normalized by dividing each weighted

Algorithm 12: Weighted Shannon entropy permutation relative importance evaluation (PRIME)

```
Data: List of probability distribution rankings from the permutations with
      (probability(p), frequency(freq))
Result: Total Weighted Entropy
WeightedProbabilities ← [];
WeightedEntropies ← [];
for each (p, freq) in Rankings do
    $wp \leftarrow p \times freq$;
    Append wp to WeightedProbabilities;
end
TotalWeighted ← sum(WeightedProbabilities);
Normalize WeightedProbabilities by TotalWeighted;
for each wp in normalized WeightedProbabilities do
    $H \leftarrow -wp \times \log_2(wp)$;
    Append H to WeightedEntropies;
end
TotalWeightedEntropy ← sum(WeightedEntropies);
Return TotalWeightedEntropy for feature importance permutations;
```

probability by the sum of these weighted probabilities to ensure that the total sums up to 1. Finally, the total weighted Shannon entropy for each normalized probability is computed by summing the individual entropies. The weighted Shannon entropy approach calculates the entropy for each individual feature and then sums these entropies to get a total measure of uncertainty across all rankings.

A higher Shannon entropy value indicates higher uncertainty or a more evenly distributed importance across features rather than a prediction of a more certain and consistent feature importance ranking.

Example 1 presents the process of computing the total weighted Shannon entropy method across all permutations. Assume the following feature importance values are observed for feature X1: Ranked 1 with 100%, which will be represented with the following notation (1,1), ranked 2nd with 100%, noted as (1,1), and ranked 3rd with 100%, noted as (1,1). Here in (1,1), the first 1 is the probability distribution, and the second 1 shows the frequency of observing that particular feature importance ranking (i.e., 1st importance, 2nd importance, 3rd importance).

1. Compute weighted probabilities (wp) = probability x frequency
   - $(1, 1), (1, 1), (1, 1) = 1, 1, 1$
2. Compute total weighted probability = sum(weighted probabilities)
   - $1 + 1 + 1 = 3$
3. Normalize weighted probabilities = wp / total weighted
   - $1/3 = 0.33$,
   - $1/3 = 0.33$,
   - $1/3 = 0.33$
4. Compute weighted Shannon entropies (WSE) = -wp * log2(wp). Here the weights are considered as outlined in steps 1 to 3.
   - $(-0.33) * \mathrm{log2}(-0.33) = 0.52$,
   - $(-0.33) * \mathrm{log2}(-0.33) = 0.52$,
   - $(-0.33) * \mathrm{log2}(-0.33) = 0.52$
5. Compute total weighted entropy = sum(weighted entropies)
   - $0.52 + 0.52 + 0.52 = 1.58$

Example 2 presents the following probability distribution and frequency information $(0.9, 2), (0.9, 1)$. This presents that the feature was ranked with the same ranking twice with a 90% probability distribution, and it was ranked with another ranking once with a 90% probability distribution.

1. Compute weighted probabilities (wp) = probability x frequency
   - $(0.9, 2), (0.9, 1) = 1.8, 0.9$
2. Compute total weighted probability = sum(weighted probabilities)
   - $1.8 + 0.9 = 2.7$
3. Normalize weighted probabilities = wp / total weighted
   - $1.8/2.7 = 0.66,$
   - $0.9/2.7 = 0.33,$
4. Compute weighted Shannon entropies (WSE) = -p * log2(p)
   - $(-0.66) * \log 2(-0.66) = 0.38,$
   - $(-0.33) * \log 2(-0.33) = 0.52,$
5. Compute total weighted entropy = sum(weighted entropies)
   - $0.38 + 0.52 =$ 0.91

These examples demonstrate the calculation of Weighted Shannon entropy values derived from the permutation importance evaluation. The Weighted Shannon entropy for Example 1 exceeds that of Example 2, despite the fact that in Example 1, all the rankings were observed with 100% probability distribution. This discrepancy arises because Example 1 presents three distinct feature importance rankings (first, second, and third most important), introducing greater uncertainty regarding the feature's importance ranking compared to Example 2. In Example 2, the feature is predicted as the most important with a 90% probability twice and as the second most important with a 90% probability once. This consistency results in a lower level of uncertainty and, consequently, a lower Weighted Shannon entropy value for Example 2 compared to Example 1, illustrating the rationale behind the differing entropy values.

The average weighted Shannon entropy values for the discussed examples will be:

$$\langle HW \rangle = \frac{1.58 + 0.91}{2} = 1.24$$

The average weighted Shannon entropy quantifies the overall uncertainty by reflecting the mean information level or predictability associated with determining feature importance values following permutations.

Finally, I have tested other weights as discussed in previous studies [197, 198, 199, 200, 201]. Specifically, I introduced a penalty mechanism for instances where the probability distribution was less than 1. This mechanism involved cases where the probability distribution fell below 1, indicating increased uncertainty about feature importance ranking consistency, the probability distribution was adjusted by multiplying it with a constant factor (e.g., 0.95). This adjustment slightly decreased the value of the probability distribution, consequently elevating the entropy of the ranking system. Despite conducting a wide range of experiments with varying penalty values and weights, I noticed that altering these weights or implementing the penalty-based Weighted Shannon entropy in the permutation relative importance evaluation had no significant difference.

3.5.5 SPEARMAN'S RANK CORRELATION COEFFICIENT

This section presents Spearman's rank correlation coefficient, the final evaluation analysis conducted in the scope of this dissertation.

To measure the closeness between the initial feature importance rankings and the permutation rankings, considering the shift in feature importance values following permutations, I used Spearman's rank correlation. It assesses how well the ranking order in two different lists aligns [252]. The Equation for Spearman's rank correlation coefficient is as follows:

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)} \tag{17}$$

Where d_i is the difference between the ranks of corresponding values in the two variables, n is the number of observations. The denominator $n(n^2 - 1)$ adjusts for ties and provides a normalization factor. $6 \sum d_i^2$ represents the sum of the squared differences between the ranks of corresponding values. ρ value ranges from -1 to 1. $\rho = 1$ indicates a perfect positive monotonic relationship. $\rho = 0$ indicates no monotonic relationship. $\rho = -1$ indicates a perfect negative monotonic relationship. A monotonic relationship refers to the consistent trend in the ranks of two features. Specifically, it means that as the values of one variable increase, the ranks of the corresponding values of the other variable consistently either increase or decrease. However, it does not necessarily imply a linear relationship.

In Chapter 4, the results of this dissertation are presented, applying the methodology outlined in the current chapter.

3.6 METHOD SUMMARY

This section presents the summary of the research method followed in this dissertation, detailing the work undertaken and the approach used.

This dissertation proposes the consideration of cooperative game theory (CGT) solutions to develop feature importance methods and enhance explainable artificial intelligence techniques, with a focus on explaining the contributions of individual features to model predictions. The motivation to develop these methods arises from the shortcomings of widely utilized techniques, such as Shapley additive explanations (SHAP), which face limitations in addressing specific challenges in explainable AI [26, 25, 1], necessitating advancements for better interpretation of complex models and their decision-making processes.

While some of the CGT solutions have been previously used to assess feature importance, the current work predominantly extends their applicability to different models and data types. Specifically, the enhancement and extension of the Shapley net effects approach [37] now include categorical data and classification models, thus expanding the scope of these techniques beyond their traditional confines.

Of particular novelty are the methods of Nucleolus, Shapley-Shubik, and conflicting claims (conflicting claims proportional, constrained equal awards, constrained equal losses, Talmud, and random arrival) based feature importance methods that are based on linear and logistic regression models. These approaches, despite their potential relevance and utility, have remained unexplored in the context of feature importance. By incorporating such novel methods, this dissertation not only extends the utility of CGT in machine learning but also opens new avenues for research into understanding and quantifying the significance of features in complex machine learning models.

Three experiments were conducted to test the feature importance methods, and it was observed that the performance of these methods varies depending on whether the data are independent or exhibit dependencies. All the feature importance methods tested in this study were effective when applied to data where the features were independent of one another. However, the methods that were specifically developed using CGT to reduce discrepancies in how rewards (or performance) are distributed among features faced challenges when dealing with data that had high multicollinearity. Multicollinearity refers to a situation where several features in the dataset are highly correlated with each other, making it difficult to distinguish their individual effects on the outcome. This situation highlighted the difficulties in applying several CGT solutions (e.g., Nucleolus, constrained equal awards, constrained equal losses) to determine the importance of features in machine learning models when the features influence each other significantly.

These feature importance methods that were developed have their strengths and weaknesses, make certain assumptions, and arrive at their conclusions differently. Which method is better and which one to choose? To evaluate the feature importance methods performance, some existing methods were considered. Model performance was evaluated using key features identified by explainable AI methods, following the approaches proposed by previous evaluation methodologies [196, 239]. However, the observed performances were very similar and, therefore, not helpful in evaluating the methods. The next evaluation approach considered was based on assessing the sensitivity (or lack thereof) of explanations to changes in the model or inputs when some data are permuted [237, 238]. Specifically, the permutation importance (PIMP) method [82] was used to compare and assess the explanation sensitivity of feature importance methods to permutations (when the data is randomly shuffled). The PIMP method involves shuffling the values within a single feature across different observations in the dataset to assess and analyze the statistical significance of the feature importance values after the permutations. Features that are determined to be statistically significant through this process are recommended for inclusion during the model-building phase, as they are likely to have a meaningful impact on the model's predictive performance. However, the effectiveness of PIMP in assessing feature significance proved to be inconsistent, as it labeled the same feature as statistically significant under some methods while deeming it insignificant in others. This inconsistency in evaluations introduced complexity in determining the real influence of features on a model's performance and in measuring the capability of feature importance methods to accurately identify critical features. This variation significantly undermines the transparency and reliability of methods designed to identify key predictors in complex models, affecting our ability to confidently understand and trust the methods' efficiency in identifying important features.

To address this issue and compare the feature importance methods, a novel evaluation metric called weighted Shannon entropy permutation importance evaluation (PRIME) that combines permutation tests and weighted Shannon entropy is introduced. Specifically, it measures how consistent feature importance methods are in predicting the feature importance values after data permutation. This approach is designed to compare the consistency of feature importance methods in predicting importance scores post-permutation. PRIME not only clarifies the influence of specific features on model outcomes but also establishes a quantifiable way of assessing the reliability of explainability techniques within the context of data permutations. The introduction of this metric aims to deepen the comprehension of feature importance and bolster the transparency and interpretability of machine learning models.

Overall, the aim of this methodology is to develop human-centric and efficient methods for explaining the significance of features in machine learning models. The objective of the Weighted Shannon Entropy-based permutation importance evaluation (PRIME) metric, specifically, is to assess the effectiveness of these methods.

CHAPTER 4

RESULTS

This chapter presents the experimental results of cooperative game theory-based feature importance methods described in Section 3.4 considering Seatpos [194], Adult Income [195], and Predator-prey dataset [176].

This chapter consists of the following sections:

1. Section 4.1 presents Shapley additive explanations (SHAP) and Local interpretable model agnostic explanations (LIME), widely used feature importance methods. The aim of this section is to outline the current landscape of widely used techniques for evaluating feature importance.

2. Section 4.2 introduces (Experiment 1) novel cooperative game theory-based feature importance methods within multiple regression modeling when data has interdependence and correlations, as demonstrated with the Seatpos dataset. This section presents the findings related to **Research Question 1**, as outlined in Section 1.3.

3. Section 4.3 introduces Experiment 2, which applies novel cooperative game theory-based feature importance methods to a logistic regression model using the Adult dataset. This section presents the findings related to **Research Question 2**, as outlined in Section 1.3.

4. Section 4.4 introduces Experiment 3 and illustrates the application of the cooperative game theory-based feature importance methods to study input variations that can be used to improve the design of an agent-based simulation model. This analysis addresses the **Research Question 3**, as outlined in Section 1.3.

5. Sections 4.2, 4.3 and 4.4 apply weighted Shannon entropy-based permutation importance evaluation (PRIME) metric to measure the consistency and uncertainty associated with the feature importance rankings. This addresses the **Research Question 4** outlined in Section 1.3.

6. Section 4.5 presents the core feature importance method, which failed to produce feature importance values for the three experiments analyzed.

4.1 REVIEW

This section explores the existing feature importance methods, considering the Seatpos dataset [194]. Widely used methods are presented, Shapley additive explanations (SHAP), local interpretable model agnostic explanations (LIME), and Permutation importance (PIMP), and a detailed description of these methods is presented in Section 2.2.1 of this dissertation. Recall that SHAP uses a cooperative game theory-based solution called Shapley values to measure the feature's contributions to the model performance from local and global scopes [15]. A local scope (local explanations) focuses on the contribution of features to individual predictions (single observation), offering detailed insights into how each feature affects a specific outcome within the model. A global scope (global explanations) evaluates the overall importance of features across all predictions made by the model, providing a comprehensive view of how each feature influences the model's performance on a broader scale. The permutation importance (PIMP), as outlined in Section 2.2.1, leverages numerous permutations of the outcome vector to ascertain the distribution of importance scores for each feature. Subsequently, PIMP utilizes the observed importance's P-value to refine the assessment of feature importance. This is achieved by constructing models that include only those features identified as statistically significant. Models with higher performance metrics underscore the ability of the feature importance method to accurately assess the significance of features.

The analysis and the results of these reviewed methods (SHAP, LIME, PIMP) are presented in the sections below using the Seatpos dataset. Extending these reviewed methods to the other datasets was deemed not necessary for achieving the research objectives.

4.1.1 SHAP: LOCAL EXPLAINABILITY

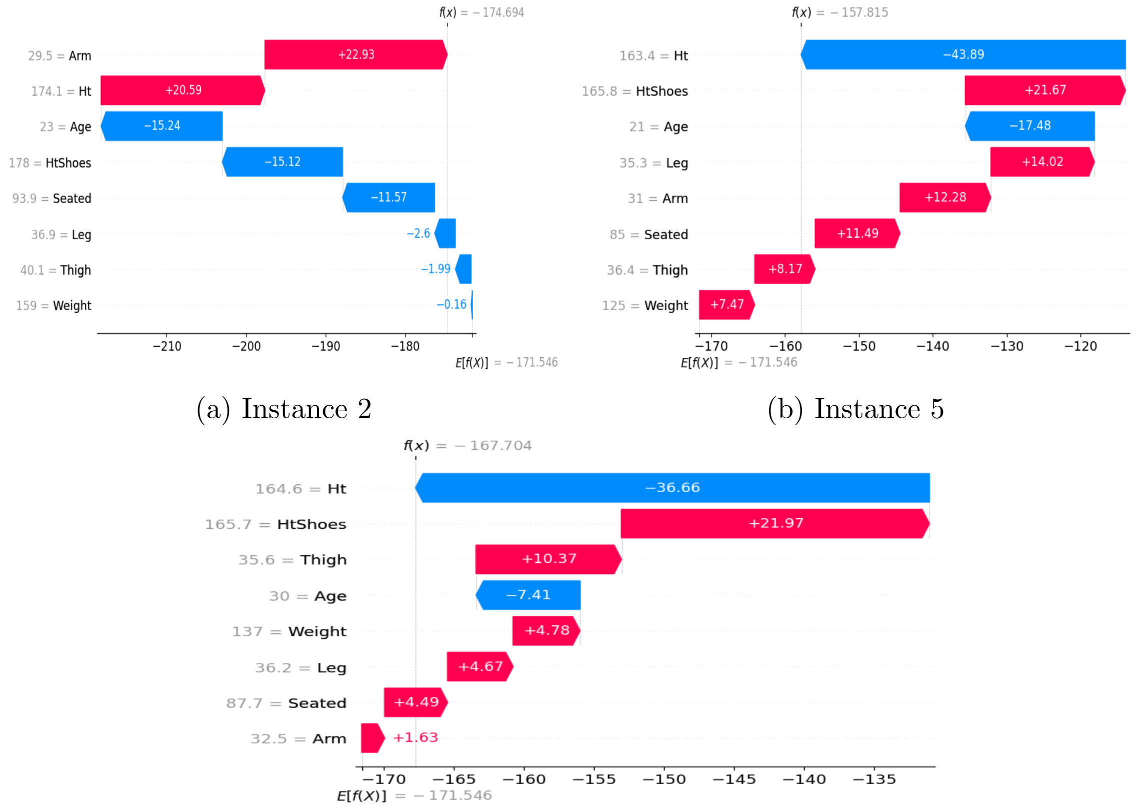
In this section, I introduce the results derived from employing SHAP local explanations on the Seatops dataset. The Seatpos dataset is subsequently used in Experiment 1, wherein the feature importance methodologies developed within the framework of this dissertation are applied.

First, I have used the Seatpos dataset to evaluate the features' individual contributions (local explanations) to a model's prediction for specific instances (Figure 11). I have looked at the outcomes of neighboring instances (instances 2, 5, and 6) to highlight the dynamic variations in feature contributions across these instances. To illustrate, consider instance 2: the most important contributor is the Arm feature, underscoring its pivotal role. Remarkably, the scenario alters, for instance, 6, where the Arm emerges as a lesser influencer. Furthermore, examining instance 2, the Ht feature is the second most influential, positively impacting the contribution. In contrast, instances 5 and 6 emphasize the significance of Ht as the primary influencer, yet with a contrary, negative effect on the prediction.

This analysis aids in explaining the dynamics that underlie the model's predictions. By dissecting the varying roles that individual features play across diverse instances, we gain deeper insights into the patterns and interplays within data. This holds particular significance in elucidating the reasoning behind individual predictions, detecting anomalies, and tackling issues tied to model fairness and bias.

Moving forward, the SHAP force plots are presented, drawing a parallel to the SHAP waterfall plots (Figure 12). Like the waterfall plots, the SHAP force plot showcases identical instances, resulting in analogous outcomes as observed in the Waterfall plots. However, this offers an alternative avenue to communicate the same insightful explanation to the user effectively.

In the force plot, features associated with SHAP values that push the model toward



(a) Instance 2

(b) Instance 5

(c) Instance 6

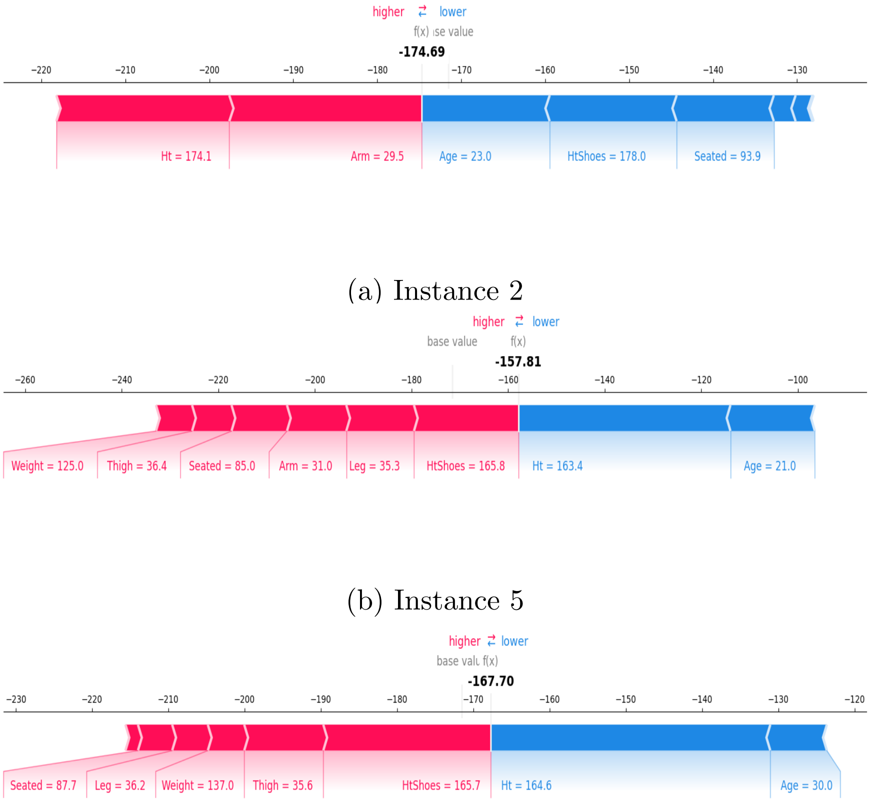
FIG. 11: SHAP waterfall plot: local explanations for various instances

increased hipcenter values are displayed in red on the left side, while those driving the model toward lower hipcenter values are depicted in blue on the right side (Figure 12). Alongside the feature names, the specific feature values are presented. Features with more significant SHAP values are depicted with longer arrow lengths.

4.1.2 SHAP: GLOBAL EXPLAINABILITY

This section presents the results obtained from the SHAP global explainability analysis.

To obtain the global explainability based on the entire dataset using the SHAP approach, I looked at the bar plot of mean absolute SHAP values for each feature. Figure 13-a presents that Ht is the most influential variable. By contrast, the least informative variable is Weight. This outcome is intuitive, as the height (Ht) can directly impact hip center positioning in car seats, as individuals with varying heights might require adjustments to



(a) Instance 2

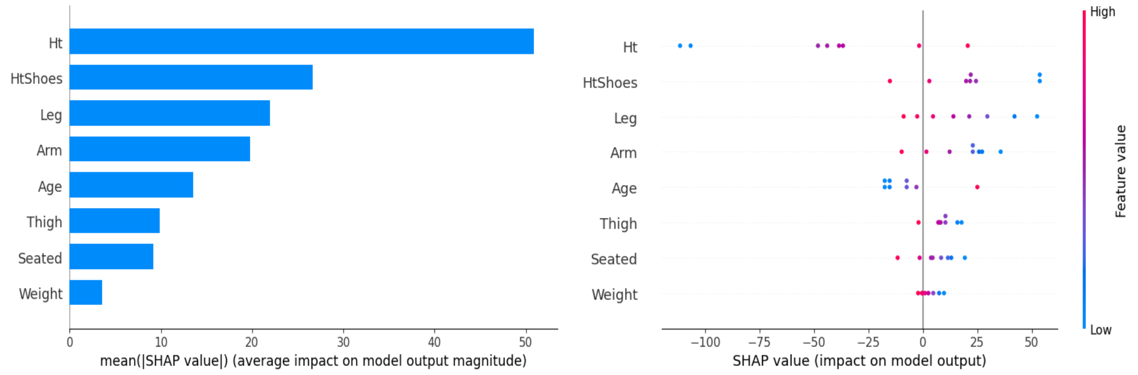
(b) Instance 5

(c) Instance 6

FIG. 12: SHAP force plot: explanations for various instances

ensure proper alignment and comfort. Taller individuals might need different seat contours or adjustments to accommodate their hip center, potentially making Ht a crucial factor. On the other hand, Weight might have a lesser impact on hip center positioning. While weight distribution can influence seating comfort, it might not play as significant a role as height when determining the optimal position for the hip center in car seats.

Next, I look at the besswarm plot where the feature ranking is exactly the same as for the bar plot (Figure 13-b). With a beeswarm plot, the underlying values of each feature can be observed related to the model's predictions. For example, we can notice that the lower values of HtShoes correspond with positive SHAP values, while Ht, which has a perfect correlation with HtShoes (with correlation coefficient = 1), its lower values correspond to negative SHAP values. This indicates that while these two features move together in lockstep,



(a) Bar plot of Mean absolute SHAP values for each feature

(b) Beeswarm plot, ranked by mean absolute SHAP value

FIG. 13: SHAP global explanations of feature importance value

additional factors beyond their correlation influence their individual impacts on the model's output. These factors might include interactions with other features or the inherent non-linearity of the model. Consequently, even though features may exhibit high correlation, their unique contributions to the model's predictions can be drastically different, leading to divergent effects on the model's overall performance and behavior. The feature Leg wields significant influence, displaying negative to nearly negligible SHAP values when values are high while exhibiting relatively higher SHAP values for low values. This underscores the impact of outliers and extreme data points on the model's predictions, emphasizing the need for more samples to enhance prediction accuracy and generalizability.

Further, the SHAP heatmap is analyzed. This provides insight into the collective influence of features on model predictions across varied instances, each accompanied by their respective SHAP values (Figure 14). My prior analysis underscored feature Ht as the paramount contributor, exerting a pronounced negative effect on the model's output, substantiated by its substantial SHAP value. Additionally, instances 0, 1 and 2 exhibited a robust positive impact on the model's prediction with features HtShoes and Leg.

Finally, I have examined the SHAP dependence plots (Figure 15). These plots reveal a consistent linear relationship across the entire range of features. Notably, the Ht feature exhibits a positive correlation with its corresponding SHAP values. The highest SHAP value

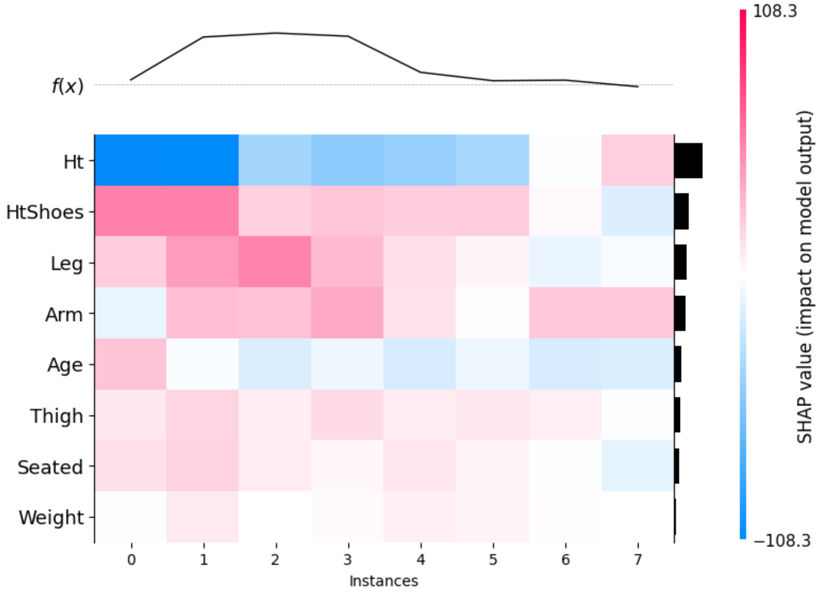


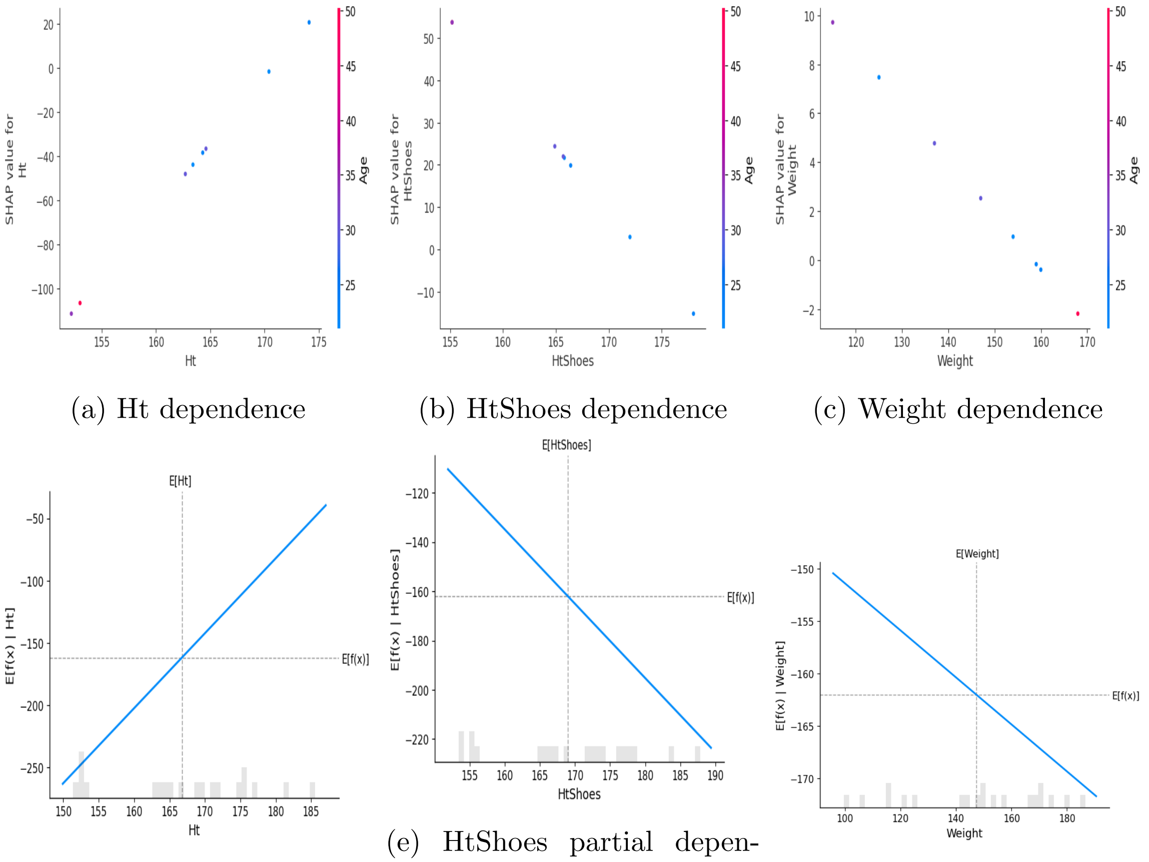
FIG. 14: SHAP heatmap

is obtained when the driver's height (Ht) measures around 175cm and their age falls within the 20-30 range. Additionally, when the age is approximately 35 and the height is below 155, a SHAP value of -100 is observed, indicating a significant negative influence on the model's prediction.

The second most important feature suggested by SHAP analysis, where Figure 15-b unveils a negative association between the height of a driver in shoes (HtShoes) and its SHAP values. Individuals with a height of approximately 155cm and an age range of 35-40 exhibit the highest SHAP values.

Next, Figure 15-c demonstrates an inverse relationship between Weight and its SHAP values, and further, it presents that the highest SHAP value is evident when the age falls within the 35-40 range, and the weight remains below 120. Conversely, the lowest SHAP value corresponds to instances where the weight is approximately 160, and the age hovers around 25 years old.

In summary, the SHAP analysis yields the following conclusion. Local explanations vary, and different sets of important features are selected for the prediction; this suggests a nuanced and context-dependent relationship between the model's output and specific input



(a) Ht dependence
(b) HtShoes dependence
(c) Weight dependence
(d) Ht partial dependence
(e) HtShoes partial dependence
(f) Weight partial dependence

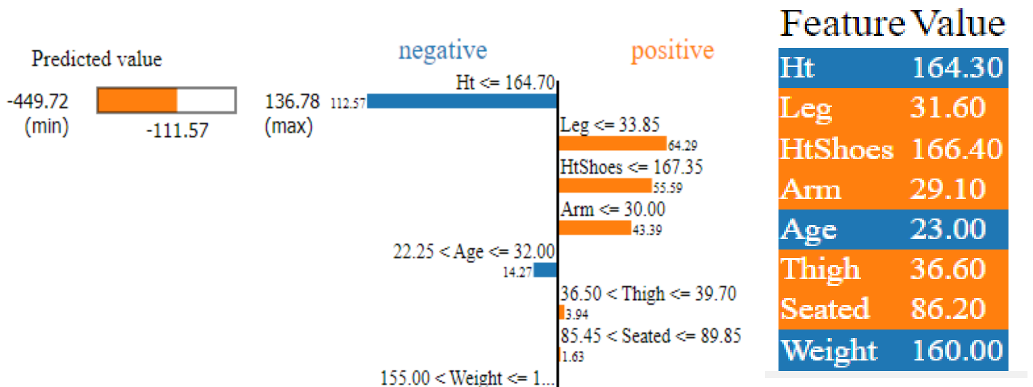
FIG. 15: SHAP dependence and partial dependence plots for the top 2 most important and the least important features, determined by mean absolute SHAP value

characteristics (Figures 11 and 12). The global explanations show that the height of an individual is an important factor respectively for predicting the hipcenter and the car seat design. The feature Weight has the least importance for the prediction. Dependence plots specifically have shown that individuals with substantially varying height values (155cm and 175cm) exhibit the highest SHAP values, exerting a substantial influence on the prediction.

4.1.3 LIME: LOCAL EXPLAINABILITY

In this section, I explored another prominent technique in explainable artificial intelligence, known as Local Interpretable Model-Agnostic Explanations (LIME), with its background detailed in Section 2.2.1). LIME focuses on explaining individual predictions (single observations or instances) around the vicinity of the specific instance being explained.

The LIME analysis presents the feature contributions of a random instance prediction (Figure 16). Figure 16-a outlines the feature contributions, while Figure 16-b illustrates the dataset's instance values. Here, Ht is the most influential feature in the prediction of this particular instance. This observation has also been corroborated by the SHAP analysis. LIME attributes the length of the leg (Leg) as the second most influential feature for the prediction, followed by HtShoes as the third significant factor. LIME designates Weight as the least influential contributor to the prediction.



Feature	Value
Ht	164.30
Leg	31.60
HtShoes	166.40
Arm	29.10
Age	23.00
Thigh	36.60
Seated	86.20
Weight	160.00

(a) LIME values for a random instance

(b) Random instance

FIG. 16: LIME: local explanations of feature importance value

LIME can also be used to show the feature importance values for ranges, as shown in the following bar plot (Figure 18). This plot presents the span of local interpretability predictions for a specified instance where the height (ht) is less than or equal to 164.7cm, height in shoes (HtShoes) is less than or equal to 167.35cm, and Age is greater than 47 and varying ranges of values for the remaining features. In this plot, the length of each bar corresponds to the importance of the respective features, offering insights into their impact on the prediction. In line with the SHAP analysis, Ht and HtShoes emerge as the most significant features for the prediction. Age emerges as the third significant feature, positively influencing the prediction.

This comprehensive analysis highlights the significance of specific features in influencing the predictions made by the model. The analysis emphasizes the observed variations in

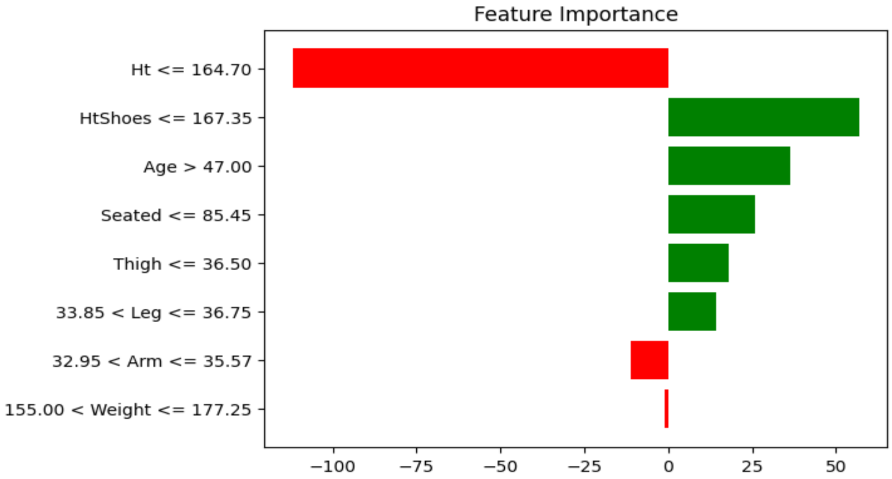


FIG. 17: LIME: range of local interpretability prediction

feature importance across different instances, unveiling the complexity and sensitivity of the model's predictive performance.

In the following sections, I delve into the application of cooperative game theory-based feature importance within the contexts of linear regression, logistic regression models, and agent-based modeling. Specifically, Experiment 1 examines the novel methods introduced in this dissertation, focusing on the use of the linear regression model with the Seatpos dataset.

Permutation importance

This section discusses the results of the permutation importance (PIMP) analysis, utilizing the methodology developed by Altmann et al. [82]. PIMP provides additional confirmation that the features HtShoes and Ht exert the most substantial influence on the prediction, while the feature Arm demonstrates the most pronounced negative impact on the prediction. Interestingly, the feature Thigh is considered the least important feature for the prediction.

TABLE 5: Sample permutation importance results for feature significance

Permuted	Age	Weight	HtShoes	Ht	Seated	Arm	Thigh	Leg
Age	Not Significant	Significant	Significant	Significant	Significant	Significant	Not Significant	Significant
Arm	Not Significant	Significant	Significant	Significant	Significant	Significant	Significant	Significant
Leg	Significant	Significant	Significant	Significant	Significant	Significant	Significant	Significant
HtShoes	Significant	Significant	Significant	Significant	Significant	Significant	Not Significant	Significant
Thigh	Not Significant	Not Significant	Not Significant	Not Significant	Not Significant	Not Significant	Significant	Not Significant
Weight	Significant	Not Significant	Significant	Significant	Significant	Significant	Significant	Significant

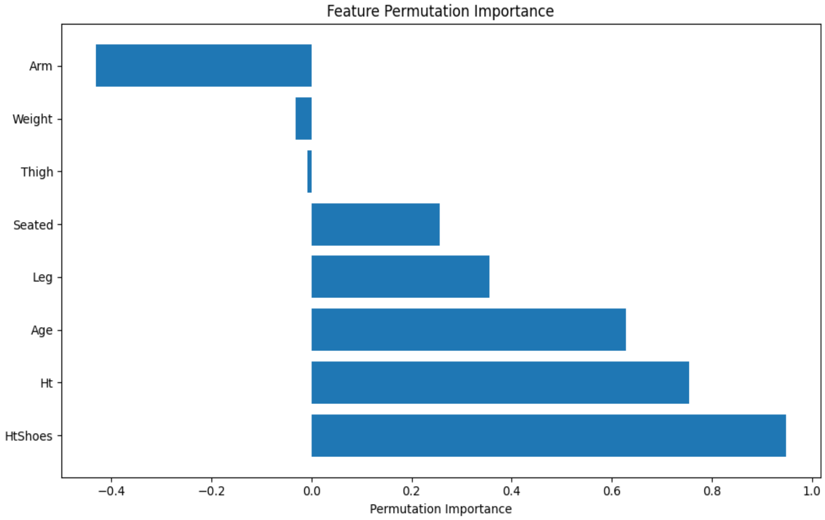


FIG. 18: Feature permutation importance

Additionally, I have documented the statistical significance of various features following the application of the PIMP method. By permuting the outcome vector, my analysis reveals that all features—with the exceptions of Weight and Thigh—hold statistical significance. Upon permuting the individual features, however, a diverse array of statistical outcomes emerged, as presented in Table 5. These outcomes showed particularly pronounced variations when examined through the different cooperative game theory-based feature importance methods. This variation posed a challenge in inclusively determining which features are truly important for the machine learning model. Further analysis of permutation importance values for different methods yields varying significance for the same feature. This underscores the necessity for a solution that can capture the performance of feature-importance methods and facilitate their comparison. The solution proposed in this dissertation is the weighted Shannon entropy-based feature importance method, which is detailed in Chapter 3, Section 3.5.

The next section presents Experiment 1, which presents various feature-importance methods applied to a linear regression model. The experiment concludes with an evaluation of these methods using the weighted Shannon entropy approach.

4.2 EXPERIMENT 1: LINEAR REGRESSION MODEL

This section discusses the cooperative-game theory-based feature importance methods in the context of a linear regression model where the features are highly correlated. This experiment considers the Seatpos dataset, where Pearson's correlation values between features range from 0.9 to 0.99. These results indicate a problem with multicollinearity, which causes inaccurate predictions and unreliable results (Table 6). Recall that Pearson's correlation is a statistic that measures the linear associations between two features [253]. The Pearson's correlation coefficient between two variables X and Y is given by $r_{XY} = \frac{\sum(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum(X_i - \bar{X})^2 \sum(Y_i - \bar{Y})^2}}$. Where r_{XY} is the Pearson's correlation coefficient between features, X and Y. X_i and Y_i are individual data points for X and Y respectively. $\bar{X}$ and $\bar{Y}$ are the mean values of X and Y, respectively. The outcome of Person's correlation ranges between (-1, 1), where -1 indicates a perfect negative correlation, 1 is a perfect positive correlation, and close to 0 shows no association between the features.

TABLE 6: Matrix of correlation coefficients and significance levels

	Age	Weight	HtShoes	Ht	Seated	Arm	Thigh	Leg
Age								
Weight	0.08							
HtShoes	-0.08	0.83***						
Ht	-0.09	0.83***	1.00***					
Seated	-0.17	0.78***	0.93***	0.93***				
Arm	0.36*	0.70***	0.75***	0.75***	0.63***			
Thigh	0.09	0.57***	0.72****	0.73***	0.61***	0.67***		
Leg	-0.04	0.78***	0.91***	0.91***	0.81***	0.75***	0.65***	
hipcenter	0.21	-0.64***	-0.80***	-0.80***	-0.73***	-0.59***	-0.59***	-0.79***

TABLE 7: Initial results of the regression model

Features	Age	Weight	HtShoes	Ht	Seated	Arm	Thigh	Leg
Estimate	0.77	0.02	-2.69	0.6	0.53	-1.32	-1.14	-6.43
P-values	0.18	0.93	0.78	0.95	0.88	0.73	0.67	0.18

Table 6 presents that feature Age is not closely associated with the rest of the features. Hipcenter seems to have a negative correlation with most of the variables, except

Age. The negative relationship is especially strong with HtShoes, Ht, and Leg. The remaining features mostly have positive associations with one another. Overall, the results show multicollinearity concerns.

Table 7 shows regression model results, which are statistical insignificance with p-values greater than 0.1. The regression coefficient for Age is 0.77, with a p-value 0.18, the coefficient for weight is 0.02, with a p-value of 0.93, HtShoes is -2.69, with a p-value of 0.78, etc. The model seems to be unsatisfactory. However, the practice and the experiments suggest that the characteristic factors of the drivers have a substantial role in predicting the drivers' car seat position [254]. The R^2 value of the model is 0.68. From these results, it is difficult to determine whether the prediction is reliable; therefore, further investigations are needed.

Next, the variance inflation factor (VIF) values were estimated to identify which features are affected by multicollinearity and the strength of the correlation. The VIF is a metric used to identify the degree of multicollinearity in a set of predictor variables within a multiple regression model [255]. VIFs start at 1 and have no upper limit. A VIF value of 1 indicates no correlation between the kth features and the remaining features and, hence, no multicollinearity. As VIF increases, it indicates greater multicollinearity. Values of VIF exceeding 5 or 10 suggest a problematic amount of multicollinearity that may need to be addressed, often leading to unstable parameter estimates, increased standard errors, and an inflated overall significance of the model [255]. Figure 19 shows the VIF for each feature.

The VIF values for Age, Weight, Arm, and Thigh are less than 5. The minimum VIF value is equal to 1.99 for Age. VIFs between 1 and 5 indicate that there is a moderate correlation. VIFs greater than 5 represent critical levels of multicollinearity where the coefficients are poorly estimated and the p-values are questionable. Features Ht and HtShoes have very large VIF values, 333 and 307, respectively. These indicators warrant corrective measures are necessary.

To address the challenges posed by multicollinearity, as indicated by elevated VIF

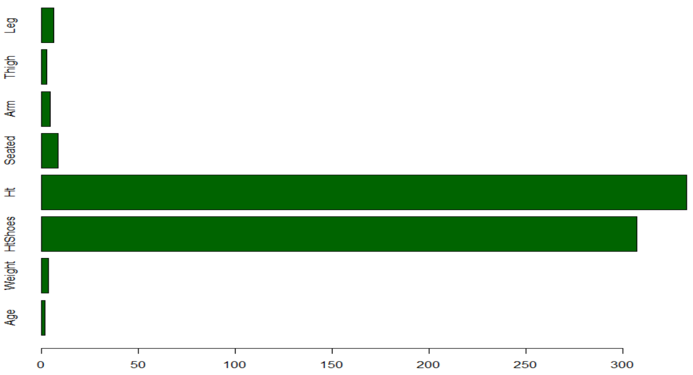


FIG. 19: Variance inflation factor (VIF) values

values, researchers might explore several strategies, such as removing features, merging variables, or applying dimensionality reduction techniques like Principal Component Analysis (PCA). However, these methods may not explain the relationship between the features and the target variable. For example, PCA is often used in the feature selection (FS) process to identify patterns in data and reduce the dimensionality [256, 257]. Feature selection refers to techniques where the objective of the algorithm is to automatically select the least subset of features that contribute the most to the model performance, manage bias-variance tradeoffs, and facilitate the model design more efficiently [258]. FS methods are broadly into three categories: filter [259], wrapper [260, 261], and embedded methods [262]. However, when it comes to the specific task of feature importance, PCA may not be the most suitable tool due to several inherent characteristics, such as loss of original feature interpretability. This is because principal components are combinations of original features, and these combinations do not directly correspond to any single feature in the dataset. This complicates the process of identifying which specific features are most important for the model's predictions, which is a key aspect of explainable AI [22]. PCA could be used as a feature reduction technique rather than a feature importance technique. This goal diverges from the aim of explainable AI, which is to understand the role and impact of individual features within the original dataset.

Feature importance methods grounded in cooperative game theory could offer deeper insights into how each feature contributes to the prediction of the model. Below, the application of various cooperative game theory-based feature importance techniques developed in this dissertation is explored. The entire dataset is used to evaluate the feature importance values, meaning that the generated insights represent the global explanations of the feature importance values.

Below, the feature importance methods are presented, followed by the Shannon entropy evaluation of these methods. One of the methods, core feature importance, which failed to produce any feature importance results, is presented in Section 4.5.

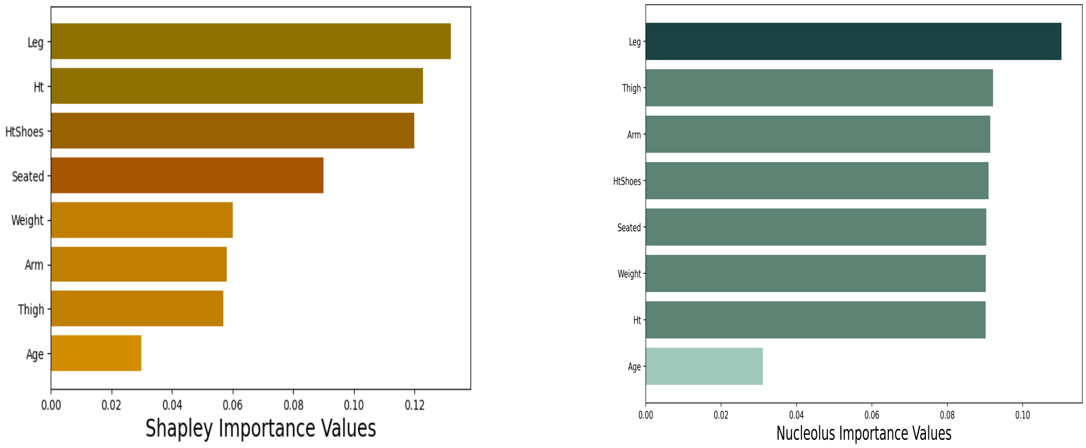
4.2.1 SHAPLEY FEATURE IMPORTANCE (SFI)

This section outlines the use of the Shapley feature importance method (SFI) to analyze the importance of feature contributions within the Seatpos dataset.

Figure 20-(a) displays the Shapley feature importance values derived from the base model for the Seatpos dataset. The base model refers to the initial model configuration before any alterations or permutations are applied to its features. SFI suggests that the feature Leg holds the top rank of importance (1), followed by height in the shoes (HtShoes), height (Ht), seated height (Seated), weight of the driver (Weight), arm length (Arm), Thigh (distance from the hip to the knee). The age of the driver is demonstrated to have minimal significance in the design of the car seat.

4.2.2 NUCLEOLUS FEATURE IMPORTANCE (NCFI)

Next, feature importance values are obtained considering Nucleolus solutions (Figure 20-(b)). Nucleolus feature importance results present that feature Leg is ranked 1st, and feature Age is ranked as the lowest important feature. The remaining features are all ranked 2nd. Nucleolus feature importance assigns the same rank to multiple features. The next rank is then adjusted accordingly. The results exhibit a reduction in specificity, stemming from the



(a) Shapley feature importance

(b) Nucleolus feature importance

FIG. 20: Shapley and Nucleolus feature importance values

tied rankings in feature importance. This lack of detail in conveying the relative significance of each variable can be disadvantageous, especially in scenarios where clear distinctions are crucial.

Adjusted Shapley and Nucleolus feature importance

It is important to note that there are several methods developed specifically to overcome the challenges associated with Shapley values, including the issue of multicollinearity among features. Basu and Maji [1] introduce an approach to tackle multicollinearity and the combined effect of features in datasets during the computation of Shapley values. The method suggests adjusting the features that are highly correlated to nullify the correlation between the feature of interest (for which the Shapley value is being calculated) and other features in the dataset. By adjusting the values of other features using these factors, the method aims to simulate a scenario where each feature is independent, thereby allowing for a more accurate computation of individual feature contributions.

The calculation of Adjustment Factors (AFs) involves several steps:

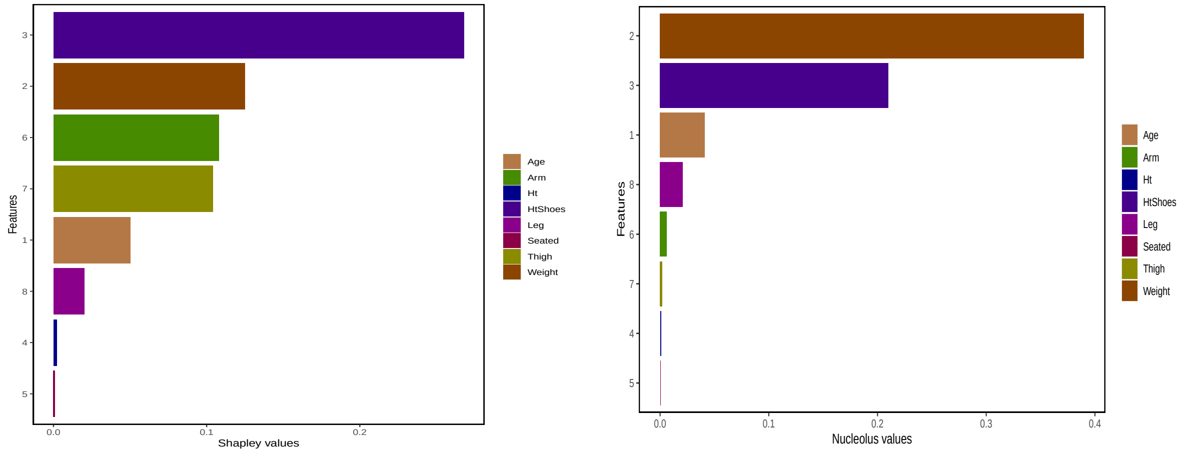
1. **Identify Correlations:** For each feature X_k not of interest (X_j), calculate the correlation between X_j and X_k.

2. **Compute Adjustment Factors:** Adjustment factors are derived such that when added to X_k, the correlation between the adjusted X_k and X_j becomes zero. This is achieved by setting:

$$AF_k = -\frac{\text{cov}(X_j, X_k)}{\text{var}(X_j)} X_j$$

effectively adjusting X_k based on the covariance between X_j and X_k, normalized by the variance of X_j.

After calculating the adjustment factors, the computation of feature importance values employs the adjusted feature values rather than the original ones.



(a) Shapley feature importance

(b) Nucleolus feature importance

FIG. 21: Shapley and Nucleolus feature importance values with adjusted factors based on Basu and Maji approach [1]

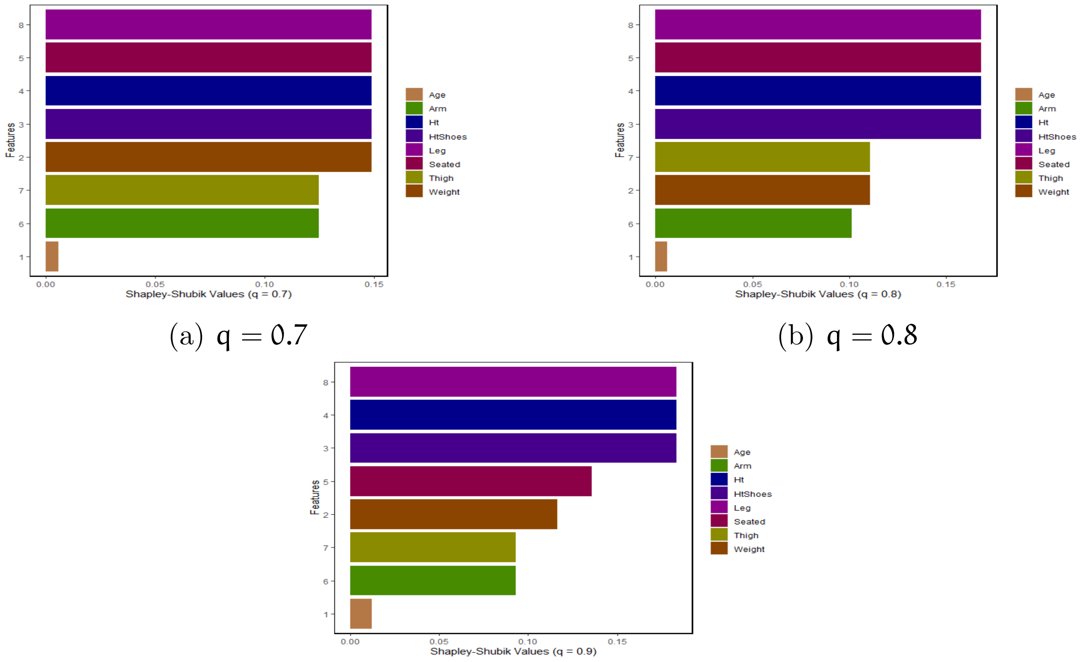
Figure 21 displays the feature importance values derived from the Shapley values and Nucleolus methods when applied to adjusted data. This adjustment effectively neutralizes the correlation among features within the Seatpos dataset.

According to these results, the two most important features for the prediction are HtShoes (Height in the shoes) and the Weight of the driver. This methodology enabled the Nucleolus feature importance to distinguish the importance of various features in contrast to scenarios involving multicollinear features. Nonetheless, this approach with adjusted factors harbors the risk of excessive adjustment, which could significantly distort the true values, potentially introducing bias. Moreover, there is a chance that these adjustments fail to accurately represent the dynamics of the relationships among features, thereby possibly injecting new biases into the calculation of feature importance values. Especially if all features are highly correlated and adjustments are made to all features to mitigate their correlations. This entails computing an adjustment factor for each feature, which could significantly deviate from the original values. However, additional experiments are necessary to fully understand how closely the feature importance values of adjusted features mirror the actual importance values.

Addressing the limitations of the Shapley value-based feature importance methods exceeds the boundaries of this dissertation's focus. Instead, I continue exploring various cooperative game theory-based techniques for feature importance evaluation that could serve as alternatives to the Shapley value method.

4.2.3 SHAPLEY-SHUBIK FEATURE IMPORTANCE (SH2FI)

This section describes the Shapley-Shubik feature importance values for the Seatpos dataset. Specifically, the figure 22-(a-c) illustrates the Shapley-Shubik values for various threshold values. The threshold value is crucial when utilizing the Shapley-Shubik index as a feature importance measure. In the Seatpos dataset, I have observed distinct feature importance values when the threshold is set high compared to when it is set low. This could occur when the features interact with each other in a complex way, and their importance values change depending on the relative weight given to each feature in the model. When the threshold is high, the contribution of each feature to the final prediction is higher, and this can affect the relative importance of the features. In this case, some features may have a higher importance value when the threshold is high compared to when it is low. However, when the threshold is low, each feature may have less impact on the final prediction, and their relative importance values may converge to a similar value. This can result in the same feature importance values being observed when the quota is small.



(a) $q = 0.7$

(b) $q = 0.8$

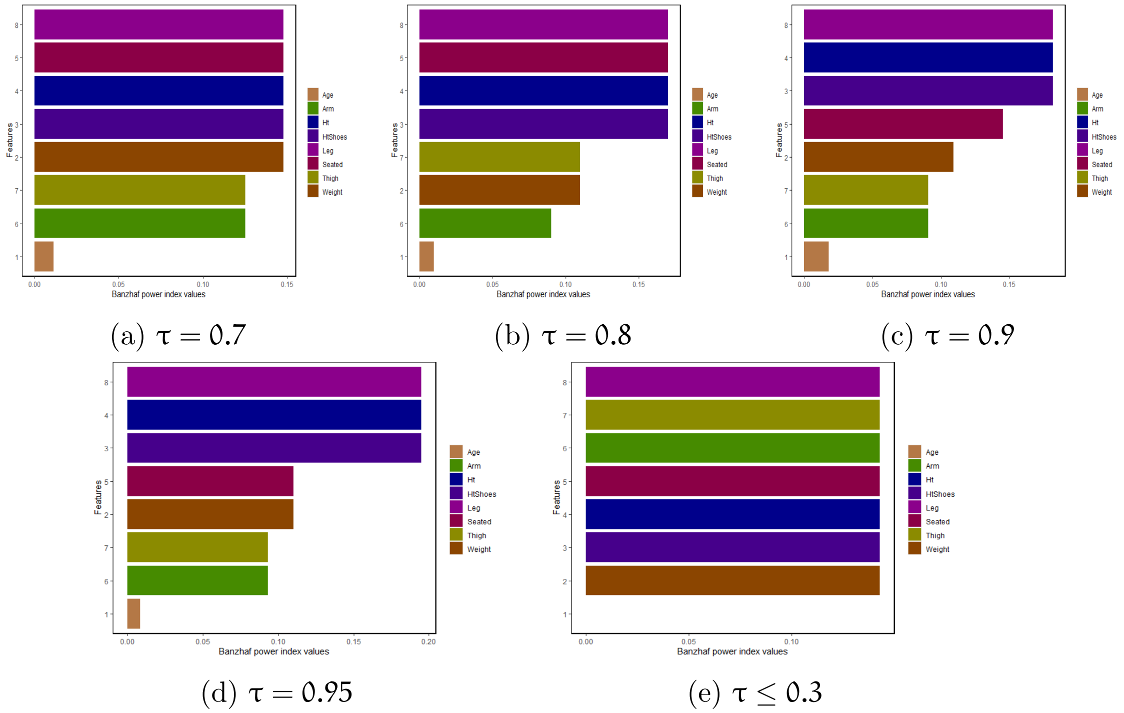
(c) $q = 0.9$

FIG. 22: Shapley-Shubik feature importance values across various thresholds

4.2.4 BANZHAF POWER INDEX FEATURE IMPORTANCE (BFI)

Several experiments were tested, altering the threshold τ value to see how the Banzhaf power feature importance values perform (see Figure 23).

Banzhaf power index calculation shows when a feature has a ”swing” vote, i.e., the power to change the model with better performance into a model with decreased performance. Figure 29 shows that the lower the threshold value, e.g., $\tau = 0.7$, there will be more critical features to achieve regression model performance equivalent to 0.7. When the threshold value increases ($\tau = 0.9$), critical features reduce. This is intuitive; however, an interesting observation can be made: one of the features (Seated) that was identified as of utmost importance for threshold values of $\tau = 0.7$ and $\tau = 0.8$ became the 4th critical important when the threshold value increased to $\tau = 0.9$. Another thing was observed is that when the threshold value is low, such as $\tau = 0.3$ or lower, the feature Age scores an importance value of 0, and the remaining features have equal critical importance (see Figure 23).



(a) $\tau = 0.7$

(b) $\tau = 0.8$

(c) $\tau = 0.9$

(d) $\tau = 0.95$

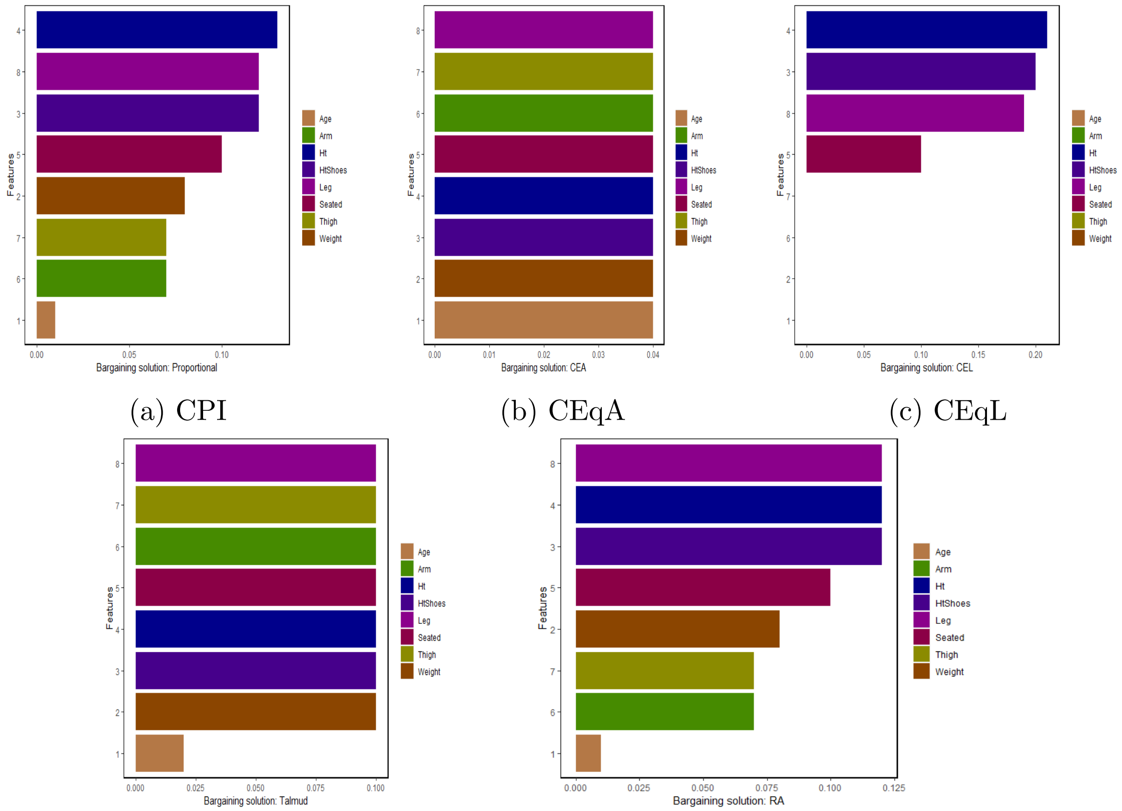
(e) $\tau \leq 0.3$

FIG. 23: Banzhaf power index feature importance values across various thresholds

4.2.5 CONFLICTING CLAIMS FEATURE IMPORTANCE VALUES

This section presents the final experiments, which analyzed the feature importance values considering conflicting claims solutions when the endowment (the desired model performance) is set to 0.7. Figures 24 presents the feature importance values for proportional (CPI), constrained equal awards (CEqA), constrained equal losses (CEqL), conflicting claims Talmud valuation (CCTV), and conflicting claims random arrival (CCRA). I tested threshold values of 0.8, 0.9, and 0.95 and found similar results to those obtained with a threshold value of 0.7.

Figure 24-(a) presents constrained proportional importance (CPI) results, where feature Ht (Height) has the highest important value. Overall, these importance values are very close to the Shapley values. The simplicity of the proportional rule further enhances its appeal, making it a valuable and practical approach for feature importance assessment. Figure 24-(b) presents that the constrained equal awards feature importance (CEqA) did not provide unique feature importance values and ranking by itself. The CEqA technique



(a) CPI

(b) CEqA

(c) CEqL

(d) CCTV

(e) CCRA

FIG. 24: Feature importance values based on conflicting claims solutions

assumes that all features have an equal claim to the target variable, which may not be the case in reality. For instance, in the Seatopos dataset, most features have equal importance values except for the age feature. However, this result could be due to multicollinearity among the features, leading to similar importance values. In such cases, the CEA technique may not accurately capture the relative importance of the features. Figure 24-(c) highlights that several features have an importance value of 0, indicating that these features do not contribute to the prediction of the target variable. This may be due to several reasons, such as high correlation with other features, lack of variation within the feature, or not being relevant to the target variable. In other words, removing this feature would not impact the model's performance. The insights derived from the CCTV analysis were not particularly useful in this case, as they indicated nearly equal feature importance values for all features, except for Age (Figure 24-(d)). This lack of differentiation among the features limits the ability to identify distinct contributions or prioritize specific factors within the context of the study. Consequently, the CCTV approach may not provide the desired granularity or discernment necessary for comprehensive feature importance analysis in this scenario. The feature importance values derived from the Random Arrival (CCRA) method demonstrate their utility by exhibiting a consistent ranking pattern, mirroring the outcomes of Shapley values, the Shapley-Shubik index, and the Banzhaf power index. This alignment suggests that the Random Arrival approach holds promise as a reliable method for measuring feature importance. Consequently, it can be considered a strong contender for assessing the relative significance of features for this problem.

4.2.6 PRIME WITH WEIGHTED SHANNON ENTROPY

This section presents the outcomes of applying the Weighted Shannon Entropies Permutation Importance Evaluation (PRIME) metric to assess the feature importance methods discussed in the scope of this experiment.

To evaluate the uncertainty of each feature's importance ranking, each feature is permuted 30 times. In total, 240 datasets were evaluated. I have also experimented with different permutation numbers (20, 50, 60); however, they all converged to the same result. The feature importance method was applied to each permutated data, and the importance values were obtained (Table 8, 9, 10, 11, 12, 13).

TABLE 8: Seatpos data: Shapley feature importance rankings after permutations

Feature	**Rank 0**	**Rank 1**	**Rank 2**	**Rank 3**	**Rank 4**	**Rank 5**	**Rank 6**	**Rank 7**	**Rank 8**
Leg	1	1 (93.3%)	1 (96.6%)	2 (50%)	2 (50%)	8 (93.3%)	1 (90%)	1 (90%)	1 (100%)
Ht	2	2 (93.3%)	2 (93.3%)	1 (50%)	1 (50%)	1 (96.6%)	2 (83.3%)	2 (90%)	2 (93.3%)
HtShoes	3	3 (96.6%)	3 (100%)	8 (90%)	8 (90%)	2 (96.6%)	3 (90%)	3 (93.3%)	3 (93.3%)
Seated	4	4 (96.6%)	4 (100%)	3 (93.3%)	3 (93.3%)	3 (100%)	8 (93.3%)	4 (96.6%)	4 (100%)
Weight	5	5 (90%)	5 (100%)	4 (93.3%)	4 (93.3%)	4 (100%)	4 (100%)	5 (96.6%)	8 (96.6%)
Arm	6	7 (86.6%)	8 (90%)	5 (83.3%)	5 (86.6%)	5 (96.6%)	5 (93.3%)	6 (96.6%)	5 (96.6%)
Thigh	7	6 (86.6%)	6 (100%)	6 (83.3%)	6 (83.3%)	6 (96.6%)	6 (93.3%)	8 (93.3%)	6 (96.6%)
Age	8	8 (86.6%)	7 (90%)	7 (90%)	7 (90%)	7 (93.3 %)	7 (93.3%)	7 (96.6%)	7 (96.6%)

Rank 0 refers to the model feature importance ranking generated by the base model, where features were not permuted. Ranks 1 through 8 describe the feature importance

TABLE 9: Seatpos data: Nucleolus feature importance rankings after permutations

Feature	Rank 0	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5	Rank 6	Rank 7	Rank 8
Leg	1	2 (60.0%)	4 (80.0%)	4 (70%)	4 (73.3%)	4 (40%)	4 (40%)	4 (73.3%)	3 (60%)
Thigh	2	4 (26.6%)	7 (33.3%)	6 (43.3%)	7 (30%)	8 (50%)	8 (40%)	5 (40%)	6 (26.6%)
Arm	2	5 (40.0%)	5 (46.6%)	7 (33.3%)	6 (53.3%)	4 (50%)	4 (50%)	5 (50%)	5 (40%)
HtShoes	2	1 (100%)	2 (96.6%)	2 (100%)	5 (53.3%)	2 (100%)	2 (100%)	2 (96.6%)	1 (100%)
Seated	2	8 (36.6%)	5 (46.6%)	8 (40.0%)	8 (46.6%)	6 (50%)	6 (50%)	7 (46.6%)	6 (36.6%)
Weight	2	7 (33.3%)	1 (100%)	1 (100%)	1 (100%)	1 (100%)	1 (100%)	1 (96.6%)	4 (33.3%)
Ht	2	6 (43%)	8 (83.3%)	5 (53.3%)	2 (100%)	8 (50%)	8 (50%)	8 (66.6%)	8 (43.3%)
Age	8	3 (60%)	3 (76.6%)	3 (73.3%)	3 (76.6%)	3 (86.6 %)	3 (86.6%)	3 (76.6%)	2 (60%)

TABLE 10: Seatpos data: SH2FI rankings after permutations

Feature	Rank 0	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5	Rank 6	Rank 7	Rank 8
Leg	1	3 (76.6%)	3 (100%)	2 (100%)	2 (100%)	8 (96.6%)	3 (100%)	3 (100%)	3 (100%)
Ht	2	1 (76.6%)	2 (100%)	8 (100%)	1 (100%)	2 (96.6%)	2 (100%)	2 (100%)	2 (100%)
HtShoes	3	2 (76.6%)	1 (100%)	1 (100%)	8 (100%)	1 (96.6%)	1 (100%)	1 (100%)	1 (100%)
Seated	4	5 (80%)	4 (100%)	3 (100%)	3 (100%)	3 (100%)	8 (100%)	4 (100%)	4 (100%)
Weight	5	4 (56.6%)	5 (100%)	4 (100%)	4 (100%)	4 (100%)	4 (100%)	5 (100%)	8 (100%)
Arm	6	6 (63.3%)	8 (100%)	5 (93.3%)	5 (100%)	5 (83.3%)	5 (100%)	6 (100%)	5 (96.6%)
Thigh	7	7 (86.6%)	6 (100%)	6 (93.3%)	6 (100%)	6 (83.3%)	6 (100%)	8 (83.3 %)	6 (96.6%)
Age	8	8 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (83.3%)	7 (100%)

TABLE 11: Seatpos data: BFI rankings after permutations

Feature	Rank 0	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5	Rank 6	Rank 7	Rank 8
Leg	1	3 (76.6%)	3 (100%)	2 (100%)	2 (100%)	8 (96.6%)	3 (100%)	3 (100%)	3 (100%)
Ht	2	1 (76.6%)	2 (100%)	8 (100%)	1 (100%)	2 (96.6%)	2 (100%)	2 (100%)	2 (100%)
HtShoes	3	2 (76.6%)	1 (100%)	1 (100%)	8 (100%)	1 (96.6%)	1 (100%)	1 (100%)	1 (100%)
Seated	4	5 (80%)	4 (100%)	3 (100%)	3 (100%)	3 (100%)	8 (100%)	4 (100%)	4 (100%)
Weight	5	4 (56.6%)	5 (100%)	4 (100%)	4 (100%)	4 (100%)	4 (100%)	5 (100%)	8 (100%)
Arm	6	6 (63.3%)	8 (100%)	5 (93.3%)	5 (100%)	5 (83.3%)	5 (100%)	6 (100%)	5 (96.6%)
Thigh	7	7 (86.6%)	6 (100%)	6 (93.3%)	6 (100%)	6 (83.3%)	6 (100%)	8 (83.3 %)	6 (96.6%)
Age	8	8 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (83.3%)	7 (100%)

TABLE 12: Seatpos data: CPI rankings after permutations

Feature	Rank 0	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5	Rank 6	Rank 7	Rank 8
Ht	1	3 (60%)	2 (56.6%)	8 (55%)	2 (65%)	2 (95%)	2 (95%)	2 (85%)	3 (70%)
Leg	2	1 (60%)	3 (56.6%)	1 (100%)	1 (65%)	8 (65%)	3 (95%)	3 (85%)	1 (70%)
HtShoes	3	2 (60%)	1 (56.6%)	2 (100%)	7 (60%)	1 (95%)	1 (95%)	1 (85%)	2 (70%)
Seated	4	4 (100%)	4 (63.3%)	3 (100%)	3 (100%)	3 (95%)	8 (65%)	4 (100%)	4 (95%)
Weight	5	5 (100%)	5 (63.3%)	4 (100%)	4 (100%)	4 (95%)	4 (100%)	5 (100%)	8 (60%)
Thigh	6	7 (93.3%)	6 (63.3%)	5 (75%)	5 (100%)	5 (75%)	6 (100%)	8 (70%)	6 (95%)
Arm	7	6 (93.3%)	8 (63.3%)	6 (75%)	6 (100%)	6 (75%)	5 (100%)	6 (100%)	5 (95%)
Age	8	8 (100%)	7 (63.3%)	6 (55%)	8 (60%)	7 (65%)	7 (65%)	7 (70%)	7 (60%)

rankings when the following features are permuted respectively: Age, Arm, Ht, HtShoes, Leg, Seated, Thigh, and Weight. Weighted Shannon entropies were evaluated using the probability distributions derived from permutation-based feature importance data. These distributions, noted in Tables 8, 9, 10, 11, 12, 13, were obtained after each permutation by

TABLE 13: Seatpos data: CCRA rankings after permutations

Feature	**Rank 0**	**Rank 1**	**Rank 2**	**Rank 3**	**Rank 4**	**Rank 5**	**Rank 6**	**Rank 7**	**Rank 8**
Leg	1	1 (70%)	1 (95%)	1 (50%)	2 (50%)	8 (70%)	1 (95%)	1 (95%)	1 (65%)
Ht	1	1 (70%)	1 (95%)	8 (60%)	1 (50%)	2 (90%)	1 (95%)	1 (95%)	1 (65%)
HtShoes	1	1 (70%)	1 (95%)	1 (50%)	7 (55%)	1 (90%)	1 (95%)	1 (95%)	1 (65%)
Seated	4	4 (100%)	4 (100%)	3 (100%)	3 (95%)	3 (100%)	8 (70%)	4 (100%)	4(100%)
Weight	5	5 (100%)	5 (100%)	4 (100%)	4 (95%)	4 (100%)	4 (100%)	5 (100%)	7 (50%)
Thigh	6	6 (70%)	6 (100%)	6 (100%)	6 (95%)	6 (100%)	6 (100%)	8 (80%)	5 (55%)
Arm	6	7 (70%)	7 (85%)	5 (100%)	5 (95%)	5 (100%)	5 (100%)	6 (100%)	6 (55%)
Age	8	8 (100%)	8 (85%)	7 (60%)	8 (55%)	7 (70%)	7 (70%)	7 (80%)	8 (55%)

observing the frequencies of the feature having a particular importance value. For example, after permutation x_1 was observed the most important feature 25 times out of 30, so its probability distribution ranked 1 (the most important feature) is $(25/30) * 100\% = 83.3\%$. This is represented in the distribution tables as $1(83.3\%)$. Next, x_2 was ranked as the 2nd most important feature 30 times out of 30 permutations, so it will be represented as $2(100\%)$. In the probability distributions, majority ranks are presented, where the rank of a feature is determined by the most frequent ranking positions across different permutation importance evaluations [263]. For example, when a feature receives the same ranking position (e.g., 8th) from multiple assessments but with varying levels of agreement (65% in one case and 35% in another), the approach defaults to the ranking that has the majority level of agreement; in this case, 65%. This means that despite the differing levels of consensus, the feature's final reported ranking is 8th, reflecting the highest percentage of agreement among the evaluations considered. This approach assumes that the most frequent or majority ranking provides a reliable estimate of a feature's relative importance under the premise that the ranking most models agree upon is likely the most accurate reflection of the feature's value. Limitations of this ranking approach are discussed in Chapter 5, Section 5.2.

Subsequently, weighted Shannon entropy values are calculated based on these feature importance ranking distributions, serving to quantify the consistency and uncertainty inherent in the feature importance rankings across various cooperative game theory-based methods developed in this dissertation. This process of computing weighted Shannon entropy-based permutation relative importance is described in Section 3.5.

I have calculated Weighted Shannon entropies for each feature within every permutation and method. Figure 25 presents PRIME weighted Shannon entropy values for Shapley feature importance, Nucleolus feature importance, Shapley-Shubik, Banzhaf power index, constrained proportional, and conflicting claims random arrival feature importance. Lower weighted Shannon entropy values suggest more consistency in the ranking of feature importance by the method. In contrast, a higher weighted Shannon entropy value indicates greater variability and more uncertainty in the method's feature importance ranking. Note that in the computation of weighted Shannon entropy permutation feature importance values, Conflicting Claims Equal Awards (CEqA), Conflicting Claims Equal Losses (CEqL), and Conflicting Claims Talmud Valuation (CCTV) were excluded from consideration. The reason for their exclusion is twofold: First, these methods displayed a lack of efficacy in predicting feature importance values within the dataset, even when permutations were applied. Second, these methods consistently predicted uniform feature importance values and corresponding rankings across all features, regardless of the permutation applied. Table 14 presents a sample example of feature importance values observed from CEqA.

TABLE 14: CEqA values when feature age is permuted

Arm - 0.12	Thigh - 0.12	Weight - 0.12	Seated - 0.12	Leg - 0.12	HtShoes - 0.12	Ht - 0.12	Age - 0.06
Arm - 0.13	Thigh - 0.13	Weight - 0.13	Seated - 0.13	Leg - 0.13	HtShoes - 0.13	Ht - 0.13	Age - 0.0
Arm - 0.1	Thigh - 0.1	Weight - 0.1	Seated - 0.1	Leg - 0.1	HtShoes - 0.1	Ht - 0.1	Age - 0.01
Arm - 0.1	Thigh - 0.1	Weight - 0.1	Seated - 0.1	Leg - 0.1	HtShoes - 0.1	Ht - 0.1	Age - 0.02

Figure 25 presents relative inconsistency and uncertainty associated with the CCRA method. Also, for some features, some methods are performing better in the feature importance ranking compared to other features. For example, SH2FI and FBI perform relatively well for the Age however, they become more inconsistent for feature Arm. For some features, the weighted Shannon entropies are low (e.g., for feature Weight, SFI, and NcFI), suggesting some level of agreement between the two methods in terms of feature importance. However, this is not consistently observed across all features. Note that the Weighted Shannon entropy values are ordered in SFI ascending order. To enhance the comparison of these methods' performances, I calculated the average Weighted Shannon entropies for each feature ranking

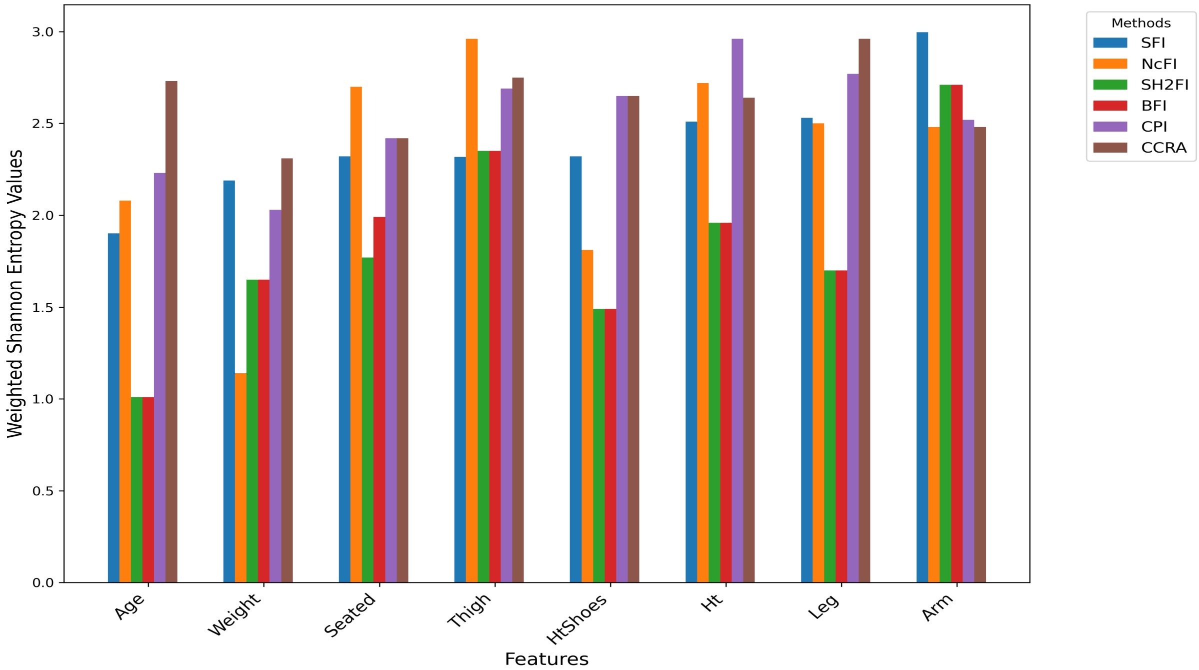


FIG. 25: Seatpos data: Weighted Shannon entropy PRIME results based on cooperative-game theory feature importance permutation methods.

provided by the methods. An example of this will be assuming WSE_1 is the Weighted Shannon entropy value for X_1, WSE_2 is the Weighted Shannon entropy value for X_2, and WSE_n is the Weighted Shannon entropy value for X_n, then the average Weighted Shannon entropy will be $\frac{WSE_1+WSE_2+WSE_n}{n}$. Figure 26 presents these average Weighted Shannon entropy values obtained from the Seatpos dataset feature importance permutation evaluations.

Figure 26 indicates that Shapley-Shubik and Banzhaf power feature importance methods have the lowest average weighted Shannon entropy values while conflicting claims random arrival exhibits the most inconsistency in feature importance ranking. These results imply that the Shapley-Shubik and Banzhaf power feature importance methods are more reliable and consistent in identifying and ranking the importance of features across various permutations. Their lower average weighted Shannon entropy values, even after data permutations, indicate that these methods more consistently agree on which features are most important, making their evaluations more reliable. On the other hand, the higher variability observed in the feature importance method based on Random Arrival (CCRA) rankings suggests it may

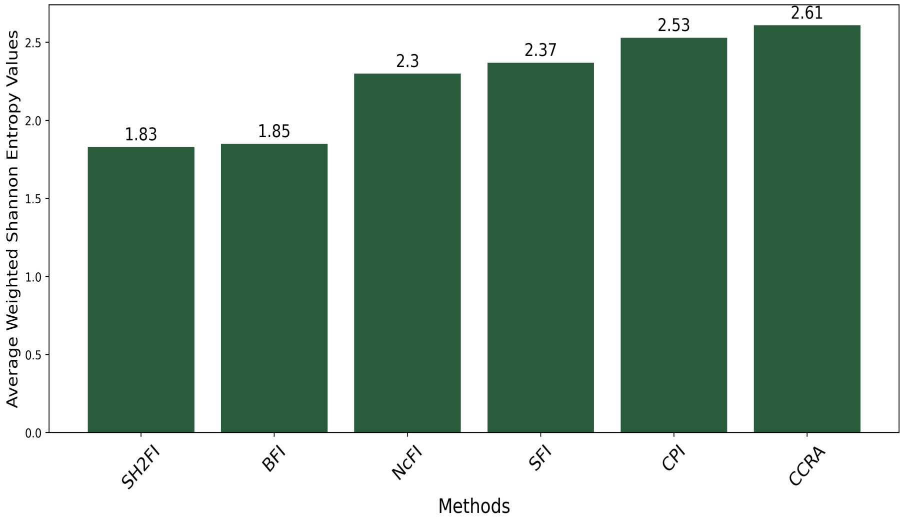


FIG. 26: Average weighted Shannon entropy PRIME results for Seatpos dataset based on cooperative game theory feature importance permutation methods.

not consistently identify the same features as important across different analyses. This inconsistency can make it less reliable for applications where consistent identification of feature importance is critical for decision-making and model interpretation.

Spearman rank correlation: Finally, the Spearman rank correlation between the original ranking and permuted feature importance ranks is measured using the data observed in Table 8, 9, 10, 11, 12, 13.

TABLE 15: Seatops data: Spearman's rank correlation coefficient between original Shapley feature importance ranking and permuted ranks

Permuted rankings	SFI	NcFI	Sh2FI	BFI	CPI	CCRA
Rank 1	0.97	0.10	0.9	0.90	0.90	0.99
Rank 2	0.92	- 0.10	0.83	0.82	0.90	0.99
Rank 3	0.61	-0.10	0.45	0.45	0.32	0.48
Rank 4	0.61	-0.10	0.61	0.61	0.73	0.62
Rank 5	0.33	-0.16	0.3	0.3	0.45	0.47
Rank 6	0.76	-0.16	0.66	0.60	0.6	0.74
Rank 7	0.97	-0.10	0.88	0.88	0.85	0.95
Rank 8	0.85	-0.10	0.76	0.76	0.76	0.91

Most of the feature importance permutation rankings generally exhibit a positive correlation with the original rankings. However, in Rank 5, when the most important feature (Leg) is permuted, the order of feature importance ranking drastically changes, resulting in the lowest correlation coefficient (0.33). Nucleolus feature importance rankings after permutations significantly differ from the original ranking, as indicated by the Spearman correlation coefficients.

In conclusion, employing permutation tests, specifically through weighted Shannon entropy, could be useful to assess the feature importance method's reliability in consistently ranking features. These metrics offer insights into the stability and robustness of the feature importance values, shedding light on the method's ability to maintain consistent rankings across different permutations.

4.3 EXPERIMENT 2: LOGISTIC REGRESSION MODEL

This section describes the experimental analysis of the application of cooperative game theory-based methods for determining feature importance, specifically within the framework of a logistic regression model utilizing the Adult dataset. From the initial dataset, which contained 11 features, only 9 informative features were used for the analysis, and the features that were not informative, such as ID, were removed. The features used to build models are marital status (MaritStat), education, occupation, age, hours per week (hours/week), sex, native country (NatCount), work class, and race. The dataset contains missing values included as ? or NaN, which I have removed prior to executing any analysis or predictive modeling. The initial data contained 48,842 observations (rows) in total, and the dataset without missing values contained 45,222 observations. Although 7.4% of the data was removed, the remaining sample size is large enough to maintain the statistical power and representativeness of the study. The minimal reduction in data does not introduce bias or adversely affect the validity of the results, providing confidence in the reliability and generalizability of the findings. This level of data retention ensures that the conclusions drawn from the analysis remain accurate and applicable to the larger population under study. The logistic regression model is used with this dataset to predict the income levels. The features within this dataset are independent and identically distributed (IID), and the data analysis shows that features do not have any correlations. The descriptions of these features and the respective categories are presented in Table 16.

TABLE 16: Description of features

Feature	**Description**
Age	The age of an individual in years
Workclass	The employment sector: * Private: Private sector employment * Local-gov: Local government employment * Self-emp-inc: Incorporated self-employment * Self-emp-not-inc: Unincorporated self-employment * State-gov: State government employment * Without-pay: No paid employment
Education	The highest level of education completed: * Education 11th: Completed up to 11th grade * Education 12th: Completed up to 12th grade * Education 7th-8th: Completed up to 7th or 8th grade * Education 9th: Completed up to 9th grade * Education Assoc-acdm: Earned an academic associate degree * Education Assoc-voc: Earned a vocational associate degree * Education Bachelors: Earned a bachelor’s degree * Education Doctorate: Earned a doctoral degree * Education HS-grad: Graduated from high school * Education Masters: Earned a master’s degree

Continued on next page

TABLE 16 – continued from previous page

Feature	Description
	* Education Preschool: Completed preschool education
	* Education Prof-school: Completed a professional school degree
	* Education Some-college: Completed some college-level education
Marital Status	The marital status of an individual:
	* Married-civ-spouse: Married to a civilian spouse
	* Married-AF-spouse: Married to an armed forces spouse
	* Married-spouse-absent: Married but currently living apart from spouse
	* Separated: Legally separated
	* Widowed: Spouse has died
	* Never-married: Never married
Occupation	The primary job or occupation of an individual:
	* Armed-Forces: Military roles
	* Craft-repair: Craftsperson and repair roles
	* Exec-managerial: Executive and managerial roles
	* Farming-fishing: Agricultural and fishing roles
	* Handlers-cleaners: Material handling and cleaning roles
	* Machine-op-inspct: Machine operation and inspection
	* Other-service: Miscellaneous service roles
	* Priv-house-serv: Private household service roles
	* Prof-specialty: Professional specialty roles
	* Protective-serv: Protective service roles, including law enforcement
	* Sales: Sales and customer service roles
	* Tech-support: Technical support roles

Continued on next page

TABLE 16 – continued from previous page

Feature	Description
Race	The ethnicity or race of an individual: * White, * Black, * Asian-Pac-Islander, * Other
Sex	The gender of an individual (Male/Female)
Hours per Week	The average number of work hours per week
Native Country	The country of origin of an individual * United-States, * Canada, * Mexico, * Other countries

Notice that this dataset is mostly described with categorical data, which often requires additional context to interpret accurately. For instance, labels like "Group A" and "Group B" are arbitrary and don't inherently convey meaning, and features with many unique categories (high cardinality) can lead to sparse data, making statistical analysis and machine learning modeling more challenging due to a lack of sufficient observations in each category. Grouping categories can sometimes lead to aggregation bias, where significant differences between subgroups within a category are overlooked, potentially obscuring important insights. For example, if a person's multiracial identity (e.g., White and Black) is not accurately reflected in the data and analysis, it can introduce biases. If individuals who identify as both White and Black can only select one option, the data misrepresents their true identity, leading to inaccurate classification and analysis. Furthermore, treating race as a single-choice attribute can result in a loss of information, masking the cultural, socioeconomic, or healthcare differences associated with a multiracial identity.

Logistic regression model results are presented in Table 17.

TABLE 17: Logistic regression model results

Features	**Estimate**	**Std. Error**	**z value**	**Pr($>|z|$)**
(Intercept)	-5.261e+00	6.816e-01	-7.718	1.18e-14 ***
age	2.899e-02	1.596e-03	18.159	$2e-16$ ***
workclass Local-gov	-6.669e-01	1.069e-01	-6.236	4.48e-10 ***
workclass Private	-4.379e-01	8.884e-02	-4.929	8.26e-07 ***
workclass Self-emp-inc	-1.888e-01	1.174e-01	-1.609	0.107627
workclass Self-emp-not-inc	-8.807e-01	1.044e-01	-8.439	2e-16 ***
workclass State-gov	-8.406e-01	1.194e-01	-7.038	1.95e-12 ***
workclass Without-pay	-1.329e+01	1.970e+02	-0.067	0.946223
education 11th	1.256e-01	2.056e-01	0.611	0.541147
education 12th	4.990e-01	2.615e-01	1.908	0.056356 .
education 7th-8th	-5.603e-01	2.353e-01	-2.381	0.017250 *
education 9th	-2.879e-01	2.620e-01	-1.099	0.271819
education Assoc-acdm	1.365e+00	1.713e-01	7.966	1.63e-15 ***
education Assoc-voc	1.346e+00	1.643e-01	8.188	2.65e-16 ***
education Bachelors	2.008e+00	1.533e-01	13.102	2e-16 ***
education Doctorate	3.095e+00	2.103e-01	14.714	2e-16 ***
education HS-grad	8.253e-01	1.492e-01	5.533	3.15e-08 ***
education Masters	2.424e+00	1.634e-01	14.839	2e-16 ***
education Preschool	-1.099e+01	1.115e+02	-0.099	0.921438
education Prof-school	3.119e+00	1.951e-01	15.989	2e-16 ***
education Some-college	1.148e+00	1.513e-01	7.586	3.30e-14 ***
marital status Married-AF-spouse	2.805e+00	4.982e-01	5.631	1.79e-08 ***
marital status Married-civ-spouse	2.076e+00	6.250e-02	33.212	2e-16 ***
marital status Married-spouse-absent	9.657e-03	2.167e-01	0.045	0.964454
marital status Never-married	-4.774e-01	7.682e-02	-6.214	5.15e-10 ***
marital status Separated	-6.828e-02	1.479e-01	-0.462	0.644287
marital status Widowed	2.191e-02	1.407e-01	0.156	0.876251
occupation Armed-Forces	-9.353e-01	1.300e+00	-0.720	0.471752

Continued on next page

TABLE 17 – continued from previous page

Variable	**Estimate**	**Std. Error**	**z value**	**Pr(z)**
occupation Craft-repair	-4.432e-04	7.535e-02	-0.006	0.995307
occupation Exec-managerial	7.865e-01	7.175e-02	10.961	2e-16 ***
occupation Farming-fishing	-1.048e+00	1.316e-01	-7.965	1.65e-15 ***
occupation Handlers-cleaners	-7.839e-01	1.378e-01	-5.687	1.29e-08 ***
occupation Machine-op-inspct	-3.509e-01	9.746e-02	-3.600	0.000318 ***
occupation Other-service	-9.204e-01	1.126e-01	-8.172	3.03e-16 ***
occupation Priv-house-serv	-2.885e+00	1.155e+00	-2.497	0.012525 *
occupation Prof-specialty	5.063e-01	7.623e-02	6.642	3.09e-11 ***
occupation Protective-serv	5.170e-01	1.205e-01	4.290	1.79e-05 ***
occupation Sales	2.538e-01	7.685e-02	3.303	0.000958 ***
occupation Tech-support	5.853e-01	1.053e-01	5.557	2.74e-08 ***
occupation Transport-moving	-1.691e-01	9.436e-02	-1.793	0.073051 .
race Asian-Pac-Islander	6.289e-01	2.643e-01	2.379	0.017350 *
race Black	4.256e-01	2.214e-01	1.922	0.054564 .
race Other	-4.804e-03	3.536e-01	-0.014	0.989159
race White	5.082e-01	2.111e-01	2.407	0.016077 *
sex Male	1.819e-01	5.049e-02	3.602	0.000316 ***
hours per week	2.965e-02	1.604e-03	18.484	2e-16 ***
native_country Canada	-9.534e-01	6.682e-01	-1.427	0.153607
native_country China	-1.933e+00	6.868e-01	-2.814	0.004889 **
native_country Columbia	-3.649e+00	1.040e+00	-3.509	0.000449 ***
native_country Cuba	-1.009e+00	6.803e-01	-1.484	0.137871
native_country Dominican-Republic	-2.376e+00	9.856e-01	-2.410	0.015934 *
native_country Ecuador	-1.629e+00	9.098e-01	-1.791	0.073366 .
native_country El-Salvador	-1.581e+00	7.628e-01	-2.073	0.038201 *
native_country England	-9.208e-01	6.824e-01	-1.349	0.177227
native_country France	-6.878e-01	8.053e-01	-0.854	0.392996
native_country Germany	-8.636e-01	6.568e-01	-1.315	0.188559

Continued on next page

TABLE 17 – continued from previous page

Variable	Estimate	Std. Error	z value	Pr(z)
native_country Greece	-2.000e+00	8.009e-01	-2.497	0.012528 *
native_country Guatemala	-1.386e+00	9.089e-01	-1.525	0.127180
native_country Haiti	-1.508e+00	8.884e-01	-1.697	0.089701 .
native_country Honduras	-2.020e+00	1.938e+00	-1.042	0.297233
native_country Hong	-1.502e+00	8.860e-01	-1.695	0.090116 .
native_country Hungary	-1.329e+00	9.529e-01	-1.395	0.163016
native_country India	-1.785e+00	6.510e-01	-2.741	0.006117 **
native_country Iran	-1.169e+00	7.257e-01	-1.610	0.107323
native_country Ireland	-8.133e-01	8.750e-01	-0.930	0.352604
native_country Italy	-5.395e-01	6.891e-01	-0.783	0.433675
native_country Jamaica	-1.416e+00	7.417e-01	-1.910	0.056191 .
native_country Japan	-1.035e+00	7.042e-01	-1.470	0.141500
native_country Laos	-2.003e+00	1.061e+00	-1.888	0.059066 .
native_country Mexico	-1.760e+00	6.464e-01	-2.722	0.006483 **
native_country Nicaragua	-1.918e+00	1.008e+00	-1.904	0.056951 .
native_country Outlying-US(Guam-USVI-etc)	-1.366e+01	2.125e+02	-0.064	0.948727
native_country Peru	-2.099e+00	1.007e+00	-2.085	0.037063 *
native_country Philippines	-1.069e+00	6.265e-01	-1.707	0.087825 .
native_country Poland	-1.390e+00	7.311e-01	-1.901	0.057244 .
native_country Portugal	-1.281e+00	8.779e-01	-1.459	0.144485
native_country Puerto-Rico	-1.500e+00	7.135e-01	-2.102	0.035529 *
native_country Scotland	-1.664e+00	1.108e+00	-1.502	0.133132
native_country South	-2.353e+00	6.984e-01	-3.369	0.000755 ***
native_country Taiwan	-1.610e+00	7.266e-01	-2.216	0.026668 *
native_country Thailand	-2.136e+00	9.831e-01	-2.173	0.029783 *
native_country Trinadad&Tobago	-1.659e+00	1.013e+00	-1.638	0.101512
native_country United-States	-1.092e+00	6.138e-01	-1.779	0.075172 .
native_country Vietnam	-2.306e+00	8.101e-01	-2.847	0.004419 **

Continued on next page

TABLE 17 – continued from previous page

Variable	Estimate	Std. Error	z value	Pr(z)
native_country Yugoslavia	-7.512e-01	8.957e-01	-0.839	0.401616

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Null deviance: 33833 on 30138 degrees of freedom

Residual deviance: 21750 on 30051 degrees of freedom

AIC: 21926

Table 17 presents the residual deviance, which drops from 33,833 (null) to 21,750 on slightly fewer degrees of freedom (from 30,138 to 30,051), which suggests that the logistic regression model with features provides a better fit than the null model. Each coefficient represents the change in the log odds of the outcome for a one-unit increase in the feature, holding all other features constant. For example, the coefficient for age (0.02899) suggests that holding all else constant, a one-year increase in age is associated with an increase in the log odds of the income being above a certain level by 0.02899. For categorical variables like workclass or education, each coefficient (e.g., workclass Local-gov = -0.6669) compares the log odds of being in that category relative to the reference category, holding other features constant. The reference feature is automatically selected by the R software based on alphabetical order or the order in which the categories appear in the dataset. For example, the coefficient (3.095, $p < 2e{-}16$) for holding a Doctorate is not only positive but also highly significant, indicating a strong association with higher odds of crossing the income threshold compared to the reference category. The very small p-value signals a high degree of confidence in this association. While the negative coefficient for ”7th-8th grade” education suggests a detrimental effect on achieving the income threshold, showing that lower education levels are associated with lower odds of high income. Overall, features like education play a critical role in income prediction. However, the degree of impact varies by education level, with higher education generally providing better odds of achieving higher income.

Overall, in predicting income levels, age emerges as a positive predictor, indicating that as individuals age, they are more likely to surpass the income threshold, possibly reflecting accumulated experience and career advancement. Marital status, particularly being married to a civilian spouse, significantly increases the likelihood of higher income, suggesting that marital stability may be associated with economic advantages. Education plays a critical role, with higher educational attainments, such as having a Bachelor's, Master's, or Doctorate degree, being strongly associated with crossing the income threshold. This underscores the value of advanced education in securing higher-paying jobs. On the other hand, lower levels of education or having no significant difference from the reference education category do not markedly increase the chances of high income, highlighting the critical threshold effect of education on earnings. Certain occupations, longer working hours, being male, and belonging to some racial backgrounds are positively associated with higher income levels. Conversely, specific occupations and being an immigrant from certain countries are negatively associated with surpassing the income threshold, highlighting the multifaceted nature of income dynamics.

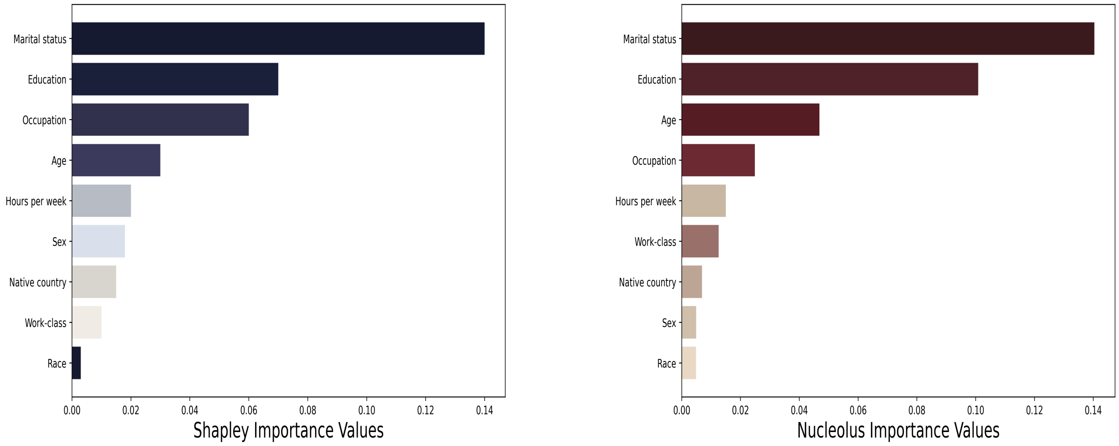
In sum, the model presents how various socio-economic features intertwine to influence income, with age, certain marital statuses, higher education, specific occupations, longer work hours, gender, race, and even native country playing significant roles in determining the likelihood of achieving higher income levels.

Below, the cooperative game theory-based explainable artificial intelligence methods are presented, followed by the Shannon entropy evaluation of these methods. One of the methods, core feature importance, which failed to produce any feature importance results, is presented in Section 4.5.

4.3.1 SHAPLEY FEATURE IMPORTANCE (SFI)

Figure 27 -(a) presents the Shapley feature importance scores of the base model. Recall the base model refers to the initial model configuration before any alterations or permutations

are applied to its features. According to the Shapley feature importance, the feature Marital status held the top rank of importance (1), followed by education, occupation age, hours per week, sex, native country, work-class, and race.



(a) Shapley feature importance

(b) Nucleolus feature importance

FIG. 27: Adult data: Shapley and Nucleolus feature importance values

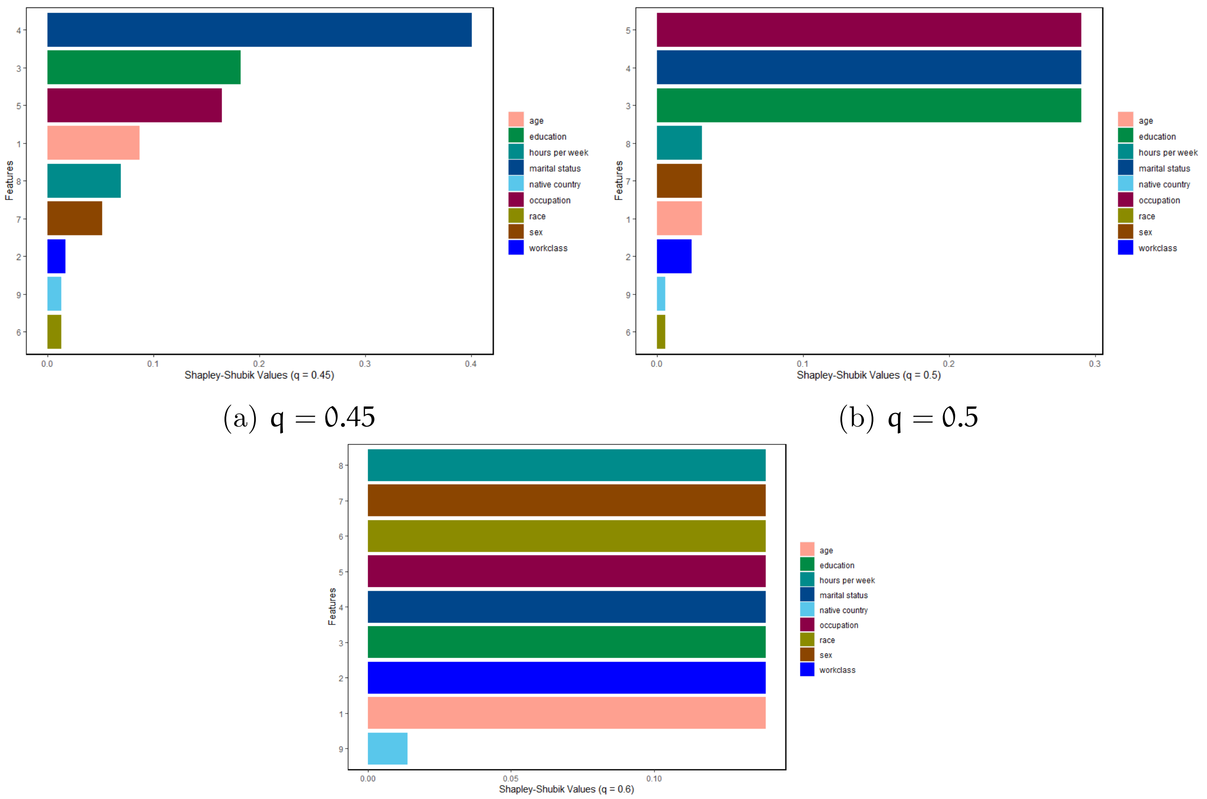
4.3.2 NUCLEOLUS FEATURE IMPORTANCE (NCFI)

The Nucleolus feature importance rankings (NcFI) for predicting income level indicate that marital status and education are the most significant factors, followed by age, occupation, hours per week, work class, native country, sex, and race in descending order of importance (Figure 27 -(b)). This implies that a person's marital status and education exhibit the most significant influence in categorizing them into higher or lower income levels. While other factors such as age, occupation, and hours worked per week also play roles, albeit to a lesser extent. Factors like native country, sex, and race are less influential in predicting income level within the dataset.

Figure 27 (a) and (b) show that Shapley and Nucleolus feature importance methods select the same set of highly important features (Marital status and Education), while the importance of the remaining features varies. The feature of least importance for predicting the income level, Race, remains the same.

4.3.3 SHAPLEY-SHUBIK FEATURE IMPORTANCE (SH2FI)

Figure 28-(a-c) shows Shapley-Shubik feature importance values. Here, as well we have observed that the feature importance values can differ depending on the threshold set. When the threshold is low, each feature may have a smaller share of the total output, leading to a more significant marginal contribution and potentially resulting in higher importance values for some features.



(a) $q = 0.45$

(b) $q = 0.5$

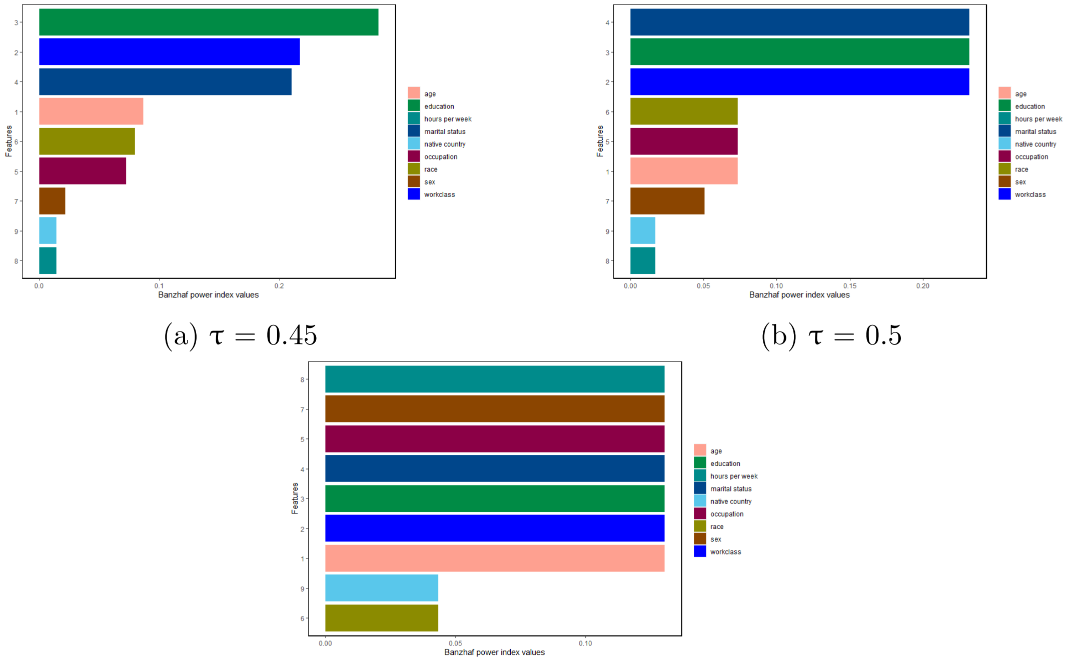
(c) $q = 0.6$

FIG. 28: Adult data: Shapley-Shubik feature importance values across various thresholds

In contrast, when the threshold is high, each feature may have a larger share of the total output, leading to less significant marginal contributions and potentially resulting in the same importance values for some features. When a threshold is set so high that even joint efforts cannot overcome it, it suggests that the feature(s) associated with that threshold are critical for the studied outcome. Despite combining the contributions of all features, they are still insufficient to reach the threshold. In this particular dataset, setting the threshold parameter (q) to 0.61 results in feature importance values of 0.11 for all features. Any value of q greater than 0.61 fails to generate feature importance values, indicating that the features, whether independently or jointly, cannot reach the threshold. This highlights the importance of the threshold value, which represents a critical point beyond which the outcome changes significantly and the associated feature(s) have a substantial impact on the outcome.

4.3.4 BANZHAF POWER INDEX FEATURE IMPORTANCE (BFI)

Similar to SH2FI, Banzhaf-power feature importance (BFI) values demonstrate their dependence on the threshold τ. A small threshold yields varying importance levels, while a high threshold requires a larger number of features to achieve it, indicating a broader set of important features. However, it is important to highlight that in the Adult dataset, both Shapley-Shubik and Banzhaf power indices fail to accurately describe feature importance values when the threshold exceeds 0.6.



(a) $\tau = 0.45$

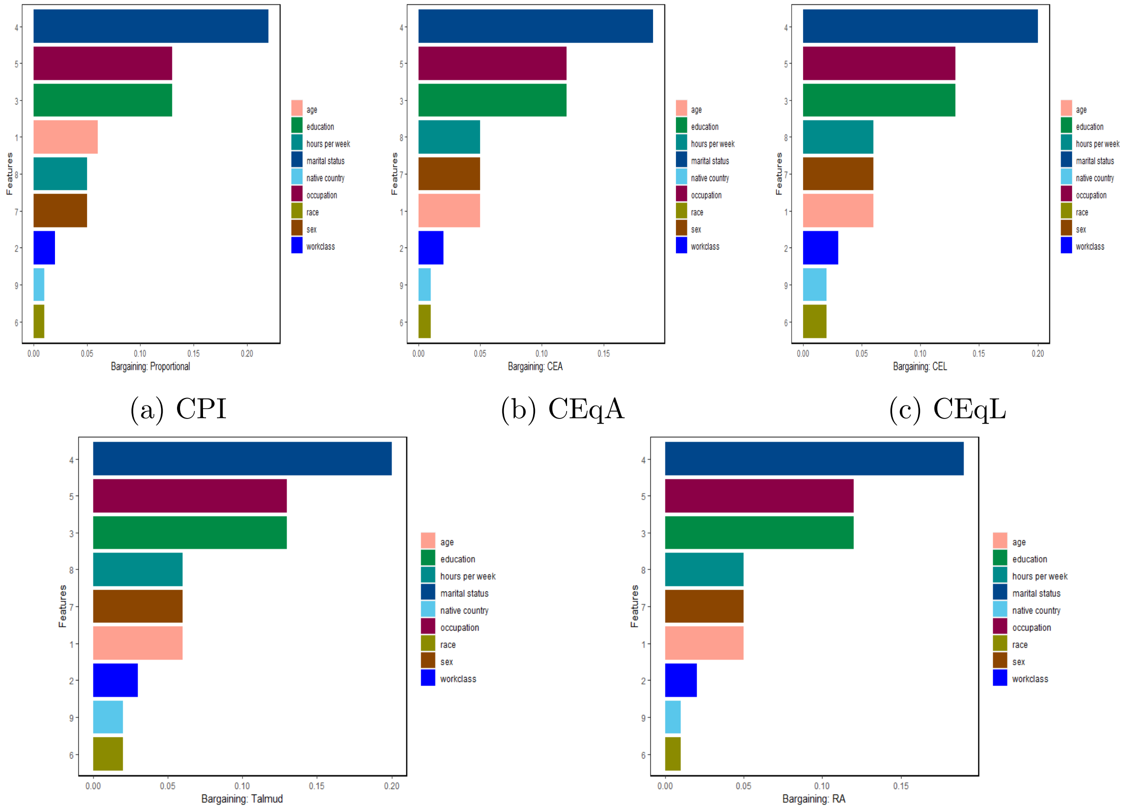
(b) $\tau = 0.5$

(c) $\tau = 0.6$

FIG. 29: Adult data: BFI values across various thresholds

4.3.5 CONFLICTING CLAIMS FEATURE IMPORTANCE VALUES

Finally, I have examined the feature importance values with bargaining solutions when the endowment (the desired model performance) is set to 0.7, presented in Figures 30. I tested threshold values of 0.8, 0.9, and 0.95, and found similar results to those obtained with a threshold value of 0.7.



(a) CPI

(b) CEqA

(c) CEqL

(d) CCTV

(e) CCRA

FIG. 30: Feature importance values based on bargaining solutions to conflicting claims

Figure 30-(a) presents the constrained proportional solution (CPI) results. According to these results, Marital status is the most important feature to predict income, followed by Occupation and Education features. Note that these features also have high correlation values with the target Figure 30. Almost identical feature importance values were observed by CEqA, CEqL, CCTV, and CCRA feature importance methods. Despite the features having equivalent claims to the resources, these features were assigned varying levels of importance rankings. Different methods uniformly concurred on this classification, suggesting a consensus across methodologies in identifying the importance values.

4.3.6 PRIME WITH WEIGHTED SHANNON ENTROPY

This section assesses the methods of feature importance by examining the consistency in the ranking of each feature's importance across the conducted permutations. A total of 60 permutations were analyzed for this evaluation. Permutation involves randomly shuffling the values of the selected feature while maintaining the values of other features. Following each permutation, the resulting feature importance values were evaluated. In total, 540 datasets were evaluated. The feature importance method was applied to each dataset, and respectively, the importance values were obtained. Table 18 presents a sample from the Shapley feature importance values after permuting feature age 60 times. Notice that the feature importance values exhibit significant changes, particularly for the permuted feature Age, while the importance values of the remaining features remain relatively consistent across permutations.

TABLE 18: Feature importance values after permuting feature Age 60 times

	Age	**Workclass**	**Education**	**Marital Status**	**Occupation**	**Race**	**Sex**	**Hours per Week**	**Native Country**
1	0.0005	0.011	0.07	0.16	0.06	0.003	0.019	0.02	0.014
2	0.0005	0.011	0.08	0.16	0.06	0.003	0.019	0.02	0.014
3	0.0008	0.011	0.08	0.16	0.06	0.003	0.019	0.02	0.014
4	0.001	0.011	0.08	0.16	0.06	0.003	0.019	0.02	0.014
...	...	...	...	...	...	...	...	...	...
59	0.00004	0.011	0.08	0.16	0.06	0.003	0.019	0.02	0.014
60	0.0003	0.011	0.08	0.16	0.06	0.003	0.019	0.02	0.014

TABLE 19: Feature importance values after permuting feature Marital status 60 times

	Age	**Workclass**	**Education**	**Marital Status**	**Occupation**	**Race**	**Sex**	**Hours per Week**	**Native Country**
1	0.05	0.01	0.07	0.001	0.07	0.003	0.071	0.02	0.015
2	0.05	0.01	0.07	0.002	0.07	0.003	0.071	0.02	0.015
3	0.054	0.01	0.07	0.001	0.07	0.003	0.071	0.02	0.015
4	0.054	0.01	0.07	0.002	0.07	0.003	0.019	0.02	0.015
...	...	...	...	...	...	...	...	...	...
59	0.054	0.01	0.07	0.001	0.07	0.003	0.032	0.02	0.015
60	0.054	0.01	0.07	0.002	0.07	0.003	0.032	0.02	0.015

Consider another sample of Shapley feature importance values obtained through the permutation of the feature Marital status (Table 19). Here again, the feature importance values demonstrate substantial variations for the permuted feature Marital status, while the importance values of the remaining features maintain a consistent trend throughout permutations.

The probability distributions were generated based on the observed feature importance rankings (feature importance values arranged in descending order), as described in Section 3.5.4. The consistency and the uncertainty of the cooperative game theory-based feature importance rankings for all the permuted features and corresponding probability distributions are presented using the data from Tables 20, 21, 22, 23, 24, 25, 26, 27, 28.

TABLE 20: Adult data: Shapley feature importance rankings after permutations

Feature	**Rank 0**	**Rank 1**	**Rank 2**	**Rank 3**	**Rank 4**	**Rank 5**	**Rank 6**	**Rank 7**	**Rank 8**	**Rank 9**
MaritStat	1	1 (100%)	1 (100%)	1 (100%)	9 (96.6%)	1 (100%)	1 (100%)	1 (100%)	1 (100%)	1 (100%)
Education	2	2 (100%)	8 (80.0%)	2 (100%)	1 (100%)	2 (100%)	2 (100%)	2 (100%)	2 (100%)	2 (100%)
Occupation	3	3 (100%)	2 (100%)	3 (100%)	2 (100%)	3 (100%)	8 (56.6%)	3 (100%)	3 (100%)	3 (100%)
Age	4	9 (100%)	3 (100%)	4 (100%)	3 (100%)	4 (100%)	3 (100%)	4 (100%)	4 (100%)	4 (100%)
Hours/week	5	4 (100%)	4 (100%)	9 (100%)	5 (100%)	5 (100%)	4 (100%)	5 (100%)	5 (100%)	5 (100%)
Sex	6	5 (100%)	5 (100%)	5 (100%)	4 (100%)	6 (100%)	5 (100%)	6 (100%)	9 (100%)	6 (100%)
NatCount	7	6 (100%)	6 (100%)	6 (100%)	6 (100%)	7 (71.6%)	6 (100%)	7 (100%)	6 (100%)	7 (100%)
Workclass	8	7 (100%)	7 (100%)	7 (100%)	7 (100%)	8 (100 %)	7 (100%)	8 (100%)	7 (100%)	9 (95%)
Race	9	8 (100%)	9 (80.0%)	8 (100%)	8 (96.6%)	9 (100%)	9 (56.6%)	9 (100%)	8 (100%)	8 (95%)

TABLE 21: Adult data Nucleolus feature importance rankings after permutations

Feature	**Rank 0**	**Rank 1**	**Rank 2**	**Rank 3**	**Rank 4**	**Rank 5**	**Rank 6**	**Rank 7**	**Rank 8**	**Rank 9**
MaritStat	1	1 (100%)	1 (100%)	1 (100%)	9 (73.3%)	1 (100%)	1 (100%)	1 (100%)	1 (100%)	1 (100%)
Education	2	2 (100%)	7 (100%)	2 (100%)	1 (100%)	2 (100%)	2 (100%)	2 (100%)	2 (100%)	2 (100%)
Age	3	9 (100%)	3 (100%)	3 (100%)	2 (100%)	3 (100%)	3 (56.6%)	3 (100%)	3 (100%)	3 (100%)
Occupation	4	3 (100%)	2 (100%)	3 (100%)	3 (100%)	4 (100%)	7 (95%)	4 (98.3%)	4 (100%)	4 (100%)
Hours/week	5	6 (100%)	5 (60%)	9 (85%)	7 (50%)	7 (68.3%)	6 (93.3%)	7 (100%)	7 (100%)	6 (100%)
Workclass	6	4 (100%)	4 (90%)	6 (98.3%)	5 (100%)	5 (83.3%)	5 (63.3%)	6 (51.6%)	5 (90%)	9 (85%)
NatCount	7	5 (100%)	6 (61.6%)	5 (98.3%)	6 (50%)	6 (51.6%)	4 (65%)	5 (51.6%)	6 (90%)	5 (100%)
Gender	8	8 (100%)	8 (71.6%)	8 (85%)	4 (100%)	9 (100 %)	9 (100%)	9 (85%)	9 (98.3%)	8 (85%)
Race	9	7 (100%)	9 (71.6%)	7 (96.6%)	8 (73.3%)	8 (100%)	8 (95%)	8 (85%)	8 (98.3%)	7 (90%)

TABLE 22: Adult data: SH2FI feature importance rankings after permutations

Feature	**Rank 0**	**Rank 1**	**Rank 2**	**Rank 3**	**Rank 4**	**Rank 5**	**Rank 6**	**Rank 7**	**Rank 8**	**Rank 9**
MaritStat	1	1 (100%)	1 (100%)	1 (100%)	9 (96.6%)	1 (100%)	1 (100%)	1 (100%)	1 (100%)	1 (100%)
Education	2	2 (100%)	8 (80.0%)	2 (100%)	1 (100%)	2 (100%)	2 (100%)	2 (100%)	2 (100%)	2 (100%)
Occupation	3	3 (100%)	2 (100%)	3 (100%)	2 (100%)	3 (100%)	8 (56.6%)	3 (100%)	3 (100%)	3 (100%)
Age	4	9 (100%)	3 (100%)	4 (100%)	3 (100%)	4 (100%)	3 (100%)	4 (100%)	4 (100%)	4 (100%)
Hours/week	5	4 (100%)	4 (100%)	9 (100%)	5 (100%)	5 (100%)	4 (100%)	5 (100%)	5 (100%)	5 (100%)
Sex	6	5 (100%)	5 (100%)	5 (100%)	4 (100%)	6 (100%)	5 (100%)	6 (100%)	9 (100%)	6 (100%)
NatCount	7	6 (100%)	6 (100%)	6 (100%)	6 (100%)	7 (71.6%)	6 (100%)	7 (100%)	6 (100%)	7 (100%)
Workclass	8	7 (100%)	7 (100%)	7 (100%)	7 (100%)	8 (100 %)	7 (100%)	8 (100%)	7 (100%)	9 (95%)
Race	9	8 (100%)	9 (80.0%)	8 (100%)	8 (96.6%)	9 (100%)	9 (56.6%)	9 (100%)	8 (100%)	8 (95%)

These probability distributions describe the consistency of each feature achieving specific ranks (See for an example Table 20). The initial rankings are labeled under the "Rank

TABLE 23: Adult data: BFI feature importance rankings after permutations

Feature	**Rank 0**	**Rank 1**	**Rank 2**	**Rank 3**	**Rank 4**	**Rank 5**	**Rank 6**	**Rank 7**	**Rank 8**	**Rank 9**
Education	1	1 (100%)	7 (80.0%)	1 (100%)	1 (100%)	2 (100%)	1 (100%)	2 (100%)	1 (100%)	1 (100%)
Workclass	2	2 (100%)	9 (100%)	2 (100%)	2 (100%)	3 (100 %)	2 (100%)	3 (100%)	3 (100%)	9 (100%)
MaritStat	3	3 (90%)	1 (100%)	3 (100%)	9 (96.6%)	1 (100%)	4 (100%)	1 (100%)	2 (100%)	2 (100%)
Age	4	9 (100%)	3 (100%)	4 (100%)	3 (100%)	4 (100%)	3 (100%)	4 (100%)	4 (100%)	4 (100%)
Race	5	4 (90%)	8 (80.0%)	8 (100%)	8 (96.6%)	9 (100%)	9 (56.6%)	9 (100%)	8 (100%)	8 (95%)
Occupation	6	5 (100%)	2 (100%)	7 (100%)	7 (100%)	8 (100%)	8 (56.6%)	6 (100%)	7 (100%)	3 (100%)
Sex	7	6 (100%)	5 (100%)	5 (100%)	4 (100%)	6 (100%)	5 (100%)	8 (100%)	9 (100%)	6 (100%)
NatCount	8	7 (100%)	6 (100%)	6 (100%)	6 (100%)	7 (71.6%)	6 (100%)	7 (100%)	6 (100%)	7 (100%)
Hours/week	9	8 (100%)	4 (100%)	9 (100%)	5 (100%)	5 (100%)	7 (100%)	5 (100%)	5 (100%)	5 (100%)

TABLE 24: Adult data: CPI feature importance rankings after permutations

Feature	**Rank 0**	**Rank 1**	**Rank 2**	**Rank 3**	**Rank 4**	**Rank 5**	**Rank 6**	**Rank 7**	**Rank 8**	**Rank 9**
MaritStat	1	1 (100%)	1 (100%)	1 (100%)	9 (96.6%)	1 (100%)	1 (100%)	1 (100%)	1 (100%)	1 (100%)
Occupation	2	3 (100%)	3 (100%)	5 (100%)	1 (100%)	3 (100%)	8 (56.6%)	3 (100%)	3 (100%)	3 (100%)
Education	3	2 (100%)	8 (90.0%)	2 (100%)	2 (100%)	2 (100%)	2 (100%)	2 (100%)	2 (100%)	2 (100%)
Age	4	9 (100%)	2 (100%)	4 (100%)	3 (100%)	4 (100%)	3 (100%)	4 (100%)	4 (100%)	4 (100%)
Hours/week	5	4 (100%)	4 (100%)	9 (100%)	5 (100%)	5 (100%)	4 (100%)	5 (100%)	5 (100%)	5 (100%)
Sex	6	5 (100%)	5 (100%)	3 (100%)	4 (100%)	6 (100%)	5 (100%)	6 (100%)	9 (100%)	6 (100%)
Workclass	7	6 (100%)	6 (100%)	6 (100%)	6 (100%)	8 (100 %)	6 (100%)	6 (100%)	6 (100%)	9 (100%)
NatCount	8	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)
Race	9	8 (100%)	9 (80.0%)	8 (100%)	8 (96.6%)	9 (100%)	9 (90%)	9 (100%)	8 (100%)	8 (95%)

TABLE 25: Adult data: CEqA feature importance rankings after permutations

Feature	**Rank 0**	**Rank 1**	**Rank 2**	**Rank 3**	**Rank 4**	**Rank 5**	**Rank 6**	**Rank 7**	**Rank 8**	**Rank 9**
MaritStat	1	1 (100%)	1 (100%)	1 (100%)	9 (96.6%)	1 (100%)	1 (100%)	1 (100%)	1 (100%)	1 (100%)
Occupation	2	2 (100%)	2 (100%)	3 (100%)	2 (100%)	3 (100%)	8 (100%)	3 (100%)	3 (100%)	3 (100%)
Education	3	3 (100%)	8 (90.0%)	2 (100%)	1 (100%)	2 (100%)	2 (100%)	2 (100%)	2 (100%)	2 (100%)
Age	4	9 (100%)	3 (100%)	4 (100%)	3 (100%)	4 (100%)	3 (100%)	4 (100%)	4 (100%)	4 (100%)
Hours/week	5	4 (100%)	4 (100%)	9 (100%)	5 (100%)	5 (100%)	4 (100%)	5 (100%)	5 (100%)	5 (100%)
Sex	6	5 (100%)	5 (100%)	5 (100%)	4 (100%)	6 (100%)	5 (100%)	6 (100%)	9 (100%)	6 (100%)
Workclass	7	6 (100%)	6 (100%)	6 (100%)	6 (100%)	8 (100 %)	6 (100%)	6 (100%)	6 (100%)	9 (100%)
NatCount	8	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)
Race	9	8 (100%)	9 (70.0%)	8 (100%)	8 (70%)	9 (100%)	9 (90%)	9 (100%)	8 (100%)	8 (95%)

TABLE 26: Adult data: CEqL feature importance rankings after permutations

Feature	**Rank 0**	**Rank 1**	**Rank 2**	**Rank 3**	**Rank 4**	**Rank 5**	**Rank 6**	**Rank 7**	**Rank 8**	**Rank 9**
MaritStat	1	1 (100%)	1 (100%)	1 (100%)	9 (100%)	1 (100%)	1 (100%)	1 (100%)	1 (100%)	1 (100%)
Occupation	2	2 (100%)	2 (100%)	3 (100%)	2 (100%)	3 (100%)	8 (100%)	3 (100%)	3 (100%)	3 (100%)
Education	3	3 (100%)	8 (90.0%)	2 (100%)	1 (100%)	2 (100%)	2 (100%)	2 (100%)	2 (100%)	2 (100%)
Age	4	9 (100%)	3 (100%)	4 (100%)	3 (100%)	4 (100%)	3 (100%)	4 (100%)	4 (100%)	4 (100%)
Hours/week	5	4 (100%)	4 (100%)	9 (100%)	5 (100%)	5 (100%)	4 (100%)	5 (100%)	5 (100%)	5 (100%)
Sex	6	5 (100%)	5 (100%)	5 (100%)	4 (100%)	6 (100%)	5 (100%)	6 (100%)	9 (100%)	6 (100%)
Workclass	7	6 (100%)	6 (100%)	6 (100%)	6 (100%)	8 (100 %)	6 (100%)	8 (100%)	6 (100%)	9 (100%)
NatCount	8	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)
Race	9	8 (100%)	9 (50.0%)	8 (100%)	8 (90%)	9 (100%)	9 (90%)	9 (100%)	8 (100%)	8 (100%)

0" column in these tables. Rank 1 presents the feature importance rankings when the feature Age is permuted. Rank 2 presents the feature importance rankings when the feature Education is permuted. Rank 3 presents the feature importance rankings when the feature Hours per Week is permuted. Rank 4 presents the feature importance rankings when the

TABLE 27: Adult data: CCTV feature importance rankings after permutations

Feature	Rank 0	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5	Rank 6	Rank 7	Rank 8	Rank 9
MaritStat	1	1 (100%)	1 (100%)	1 (100%)	9 (85%)	1 (100%)	2 (100%)	1 (100%)	1 (100%)	1 (100%)
Occupation	2	2 (100%)	2 (100%)	3 (100%)	2 (100%)	3 (100%)	8 (85%)	3 (100%)	3 (100%)	3 (100%)
Education	3	3 (100%)	8 (100%)	2 (100%)	1 (100%)	2 (100%)	1 (100%)	2 (100%)	2 (100%)	2 (100%)
Age	4	9 (100%)	3 (100%)	4 (100%)	3 (100%)	4 (100%)	3 (100%)	4 (100%)	4 (100%)	8 (100%)
Hours/week	5	4 (100%)	4 (100%)	9 (100%)	5 (100%)	5 (100%)	4 (100%)	5 (100%)	5 (100%)	5 (100%)
Sex	6	5 (100%)	5 (100%)	5 (100%)	4 (100%)	6 (100%)	5 (100%)	6 (100%)	9 (100%)	6 (100%)
Workclass	7	8 (100%)	6 (100%)	6 (100%)	6 (100%)	8 (100 %)	6 (100%)	8 (100%)	8 (100%)	9 (100%)
NatCount	8	6 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)
Race	9	7 (100%)	9 (80.0%)	8 (100%)	8 (85%)	9 (100%)	9 (90%)	9 (100%)	6 (100%)	4 (95%)

TABLE 28: Adult data: CCRA feature importance rankings after permutations

Feature	Rank 0	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5	Rank 6	Rank 7	Rank 8	Rank 9
MaritStat	1	1 (100%)	1 (100%)	1 (100%)	9 (100%)	1 (100%)	1 (100%)	1 (100%)	1 (100%)	1 (100%)
Occupation	2	2 (100%)	2 (100%)	3 (100%)	2 (100%)	3 (100%)	8 (100%)	3 (100%)	3 (100%)	3 (100%)
Education	3	3 (100%)	8 (90.0%)	2 (100%)	1 (100%)	2 (100%)	2 (100%)	2 (100%)	2 (100%)	2 (100%)
Age	4	9 (100%)	3 (100%)	4 (100%)	3 (100%)	4 (100%)	3 (100%)	4 (100%)	4 (100%)	4 (100%)
Hours/week	5	4 (100%)	4 (100%)	9 (100%)	5 (100%)	5 (100%)	4 (100%)	5 (100%)	5 (100%)	5 (100%)
Sex	6	5 (100%)	5 (100%)	5 (100%)	4 (100%)	6 (100%)	5 (100%)	6 (100%)	9 (100%)	6 (100%)
Workclass	7	6 (100%)	6 (100%)	6 (100%)	6 (100%)	8 (100 %)	6 (100%)	8 (100%)	6 (100%)	9 (100%)
NatCount	8	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (100%)	7 (85%)	7 (100%)	7 (100%)
Race	9	8 (100%)	9 (90.0%)	8 (100%)	8 (100%)	9 (100%)	9 (100%)	9 (85%)	8 (100%)	8 (100%)

feature Marital Status is permuted. Rank 5 presents the feature importance rankings when the feature Native Country is permuted. Rank 6 presents the feature importance rankings when the feature Occupation is permuted. Rank 7 presents the feature importance rankings when the feature Race is permuted. Rank 8 presents the feature importance rankings when the feature Sex is permuted. Rank 9 presents the feature importance rankings when the feature Workclass is permuted.

The feature importance rankings shift after permutation, but the overall order remains the same, which refers to the hierarchy of features based on their importance values. This can be attributed to the impact of random noise introduced by the permutation process. For example, assume feature x_1 is initially ranked the highest, followed by x_2, x_3, x_4, and x_5. After permuting x_1, x_1 appears to have the lowest importance ranking. The same happens when permuting x_2, and so on. The other features' rankings remain relatively stable. For instance, consider the original order of importance as $x_1 > x_2 > x_3 > x_4 > x_5$. After permuting x_1, it is observed $x_1 < x_2 > x_3 > x_4 > x_5$. While x_1 now has a lower importance score, the relative order of x_2, x_3, x_4, and x_5 remains the same. The outcome that the overall order of feature importance remains the same suggests that the model consistently identifies the relative importance of features.

Given these probability distributions, weighted Shannon entropy is computed for each feature importance ranking using Algorithm 12. An example computation is included in Section 4.2.6. Figure 31 presents the weighted Shannon entropies values across different methods.

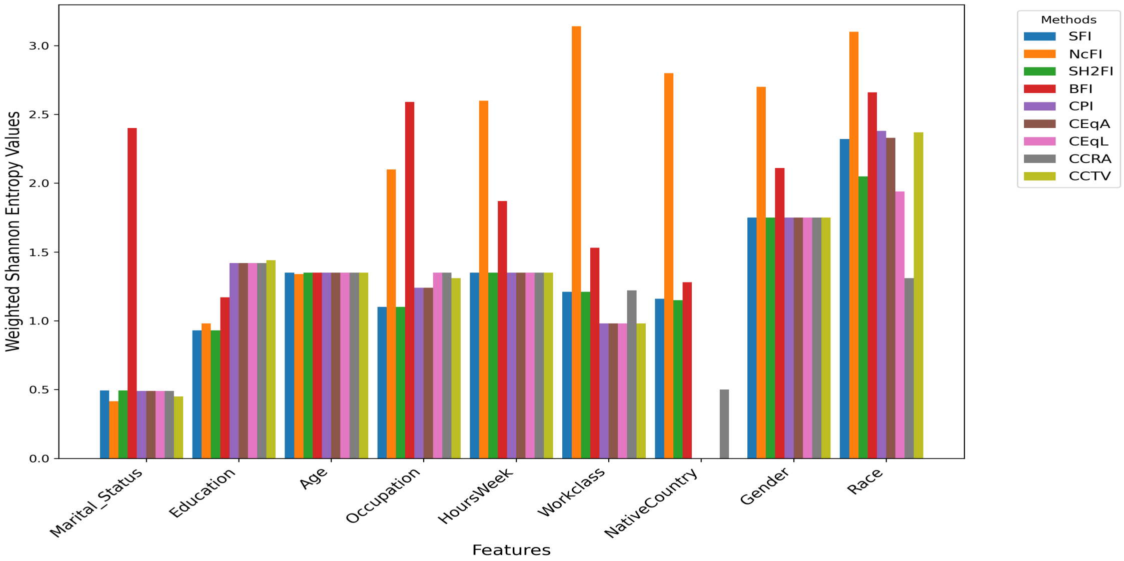


FIG. 31: Weighted Shannon entropy PRIME results based on cooperative-game theory feature importance permutation methods.

Similar to Experiment 1, certain features have relatively lower weighted Shannon entropy values, such as Marital Status, while others, such as Race, exhibit more uncertainty in feature importance ranking. This indicates that for some features, the methods easily identify their respective importance values, whereas for other features, determining these values proves to be more challenging. This could be due to more complex interactions of some features with others, making it harder to isolate their individual impact on the model's outcome. Further, in this experiment, more uniform agreement about the feature importance rankings is reached. There are some exceptions, mostly exhibited by the FBI and NcFI. These exceptions suggest moments where FBI and NcFI methods assess the importance of features differently, potentially due to their unique approach to valuing the contribution of each feature within the context of all possible feature combinations. Higher Shannon entropy values for the NcFI method further highlight the complexity and sensitivity of this approach in evaluating feature importance values. Overall, this implies that the importance assigned to each feature fluctuates more across different evaluations or data subsets, indicating a less consistent ranking of features compared to methods yielding lower entropy values.

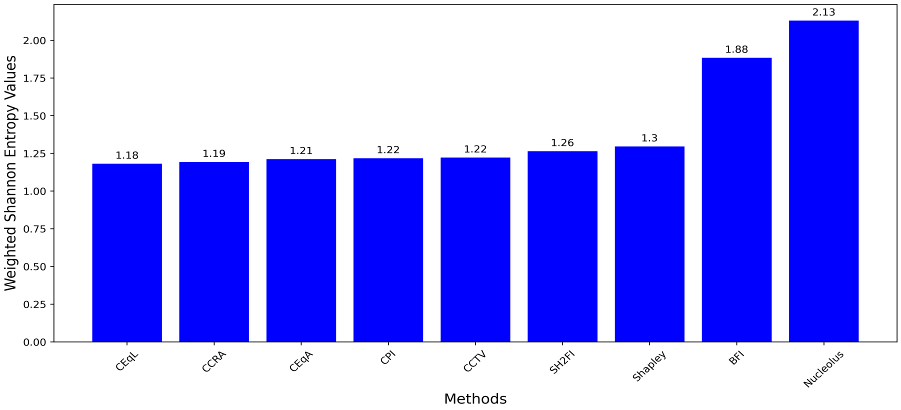


FIG. 32: Average weighted Shannon entropy PRIME results for Adult dataset based on cooperative-game theory feature importance permutation methods.

Finally, the average weighted Shannon entropy values were analyzed (Figure 32). The results show very close average Weighted Shannon entropy values across many feature importance methods, including CEqL, CCRA, CEqA, CPI, CCTV, SH2FI, and SFI. Methods of determining feature importance that utilize the Banzhaf power index and Nucleolus exhibit increased average weighted Shannon entropy PRIME values. This implies that the feature importance rankings derived from the Banzhaf power index and Nucleolus methods exhibit greater variability and unpredictability. The higher Shannon entropy PRIME values suggest that these methods may not consistently agree on the importance of features across different permuted datasets. This variability could indicate a more nuanced or sensitive approach to ranking feature importance, potentially capturing complex interactions or dependencies not as readily identified by methods with lower entropy values.

Overall, from this experiment, the weighted Shannon entropy values associated with this dataset are lower than those observed in the previous experiment with Seatpos dataset. Higher entropy values in Experiment 1 imply a greater degree of uncertainty or variability in the feature importance rankings. This could be related to the close linkage between feature importance rankings within this dataset. This interconnectedness can be traced back to the high degree of correlation among the features. When features are strongly correlated, their individual impacts on the model exhibit similarity, leading to a closely aligned ranking of importance. The correlation observed among features enhances their substitutability, enabling a scenario where multiple features can be selected for distinct rankings or several features share equal importance in prediction. Consequently, this dynamic introduces a heightened level of uncertainty into the feature importance ranking.

Spearman rank correlation: Finally, the Spearman rank correlation between the original ranking and permuted feature importance ranks is measured using the data observed in Tables 20, 21, 22, 23, 24, 25, 26, 27, 28.

TABLE 29: Adult data: Spearman's rank correlation coefficient between original and permuted ranks

Permuted rankings	SFI	NcFI	Sh2FI	BFI	CPI	CEqA	CEqL	CCTV	CCRA
Rank 1	0.73	0.58	0.75	0.75	0.73	0.75	0.75	0.70	0.75
Rank 2	0.65	0.71	0.61	- 0.23	0.71	0.75	0.70	0.70	0.75
Rank 3	0.83	0.79	0.83	0.84	0.68	0.81	0.81	0.81	0.81
Rank 4	0.38	0.25	0.38	0.36	0.38	0.36	0.36	0.36	0.36
Rank 5	1	0.93	1	0.63	0.96	0.96	0.96	0.96	0.96
Rank 6	0.75	0.81	0.75	0.71	0.65	0.65	0.65	0.61	0.65
Rank 7	1	0.91	1	0.66	0.93	0.96	0.97	0.97	0.96
Rank 8	0.9	0.93	0.9	0.7	0.88	0.88	0.88	0.88	0.88
Rank 9	0.98	0.84	0.98	0.28	0.93	0.93	0.93	0.60	0.93

Table 29 reveals the following pattern: most permutation ranks exhibit a strong correlation with the original feature importance ranking. However, an exception arises in the case where the most important feature (Marital Status) undergoes permutation. This specific permutation distorts the entire feature importance hierarchy, resulting in a comparatively low Spearman correlation coefficient. In other words, the original order of feature importance is not preserved after this particular permutation, emphasizing the model's reliance on Marital Status is not replicated by other features when Marital Status is permuted. This highlights the specific contribution of Marital Status to the model's predictive performance.

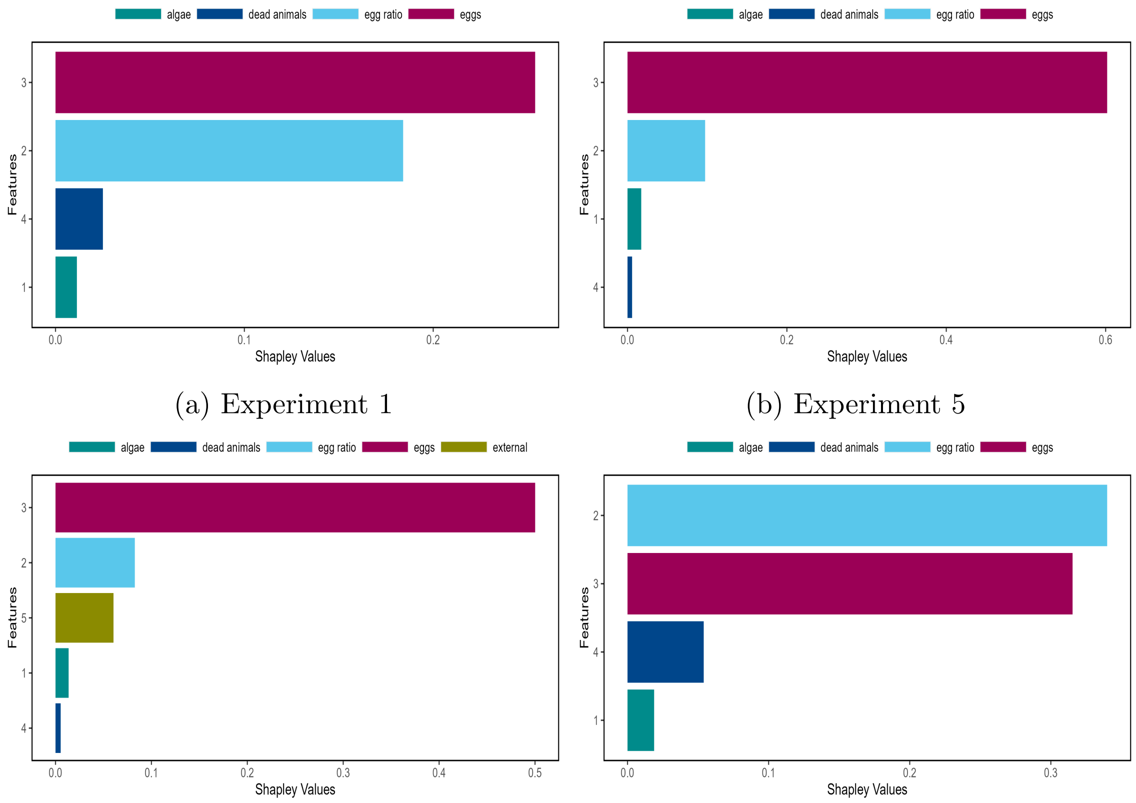
In Rank 5 and 7, the permutation of two relatively less important features (Native country and Race) produces perfect (1) and near perfect (0.93 and 0.91) feature importance ranking correlation with the initial feature importance ranking. The perfect correlation implies that the model's performance is not altered when these features are permuted, reinforcing their lower impact on the overall predictive outcome.

Furthermore, the SFI and SH2FI methods display similar correlation coefficients, indicating a strong alignment in their assessments of feature importance. Similarly, methods that are grounded in resolving conflicting claims also demonstrate correlation coefficients that are similar to one another. This pattern suggests that methods sharing a theoretical basis or approach to evaluating feature importance tend to produce similar results, underscoring the influence of the underlying principles on the outcomes of feature importance assessments. This consistency across these methods provides insights into the reliability and comparability of different feature importance evaluation strategies.

4.4 EXPERIMENT 3: INPUT DATA ANALYSIS FOR AGENT-BASED MODELING

This section discusses the third experiment, which involves applying cooperative game theory-based feature importance methods to study input data describing a predator-prey scenario, and this data was collected by Blasius et al., [176]. This application of feature importance methods with empirical input data has the potential to be useful when actual agent-based simulation models are developed. The results discussed in this section have been published in Winter Simulation Conference [177]. The description of this problem is presented in Sections 2.6 (ABM), 3.2 (data), and 3.4.4 (ABM and feature importance).

The goal of this third experiment is to illustrate the variability in feature importance values as a result of changes in inputs and parameters across 10 different experiments that were collected by Blasius et al., [176]. I have used the data from these experiments that



(a) Experiment 1

(b) Experiment 5

(c) Experiment 8

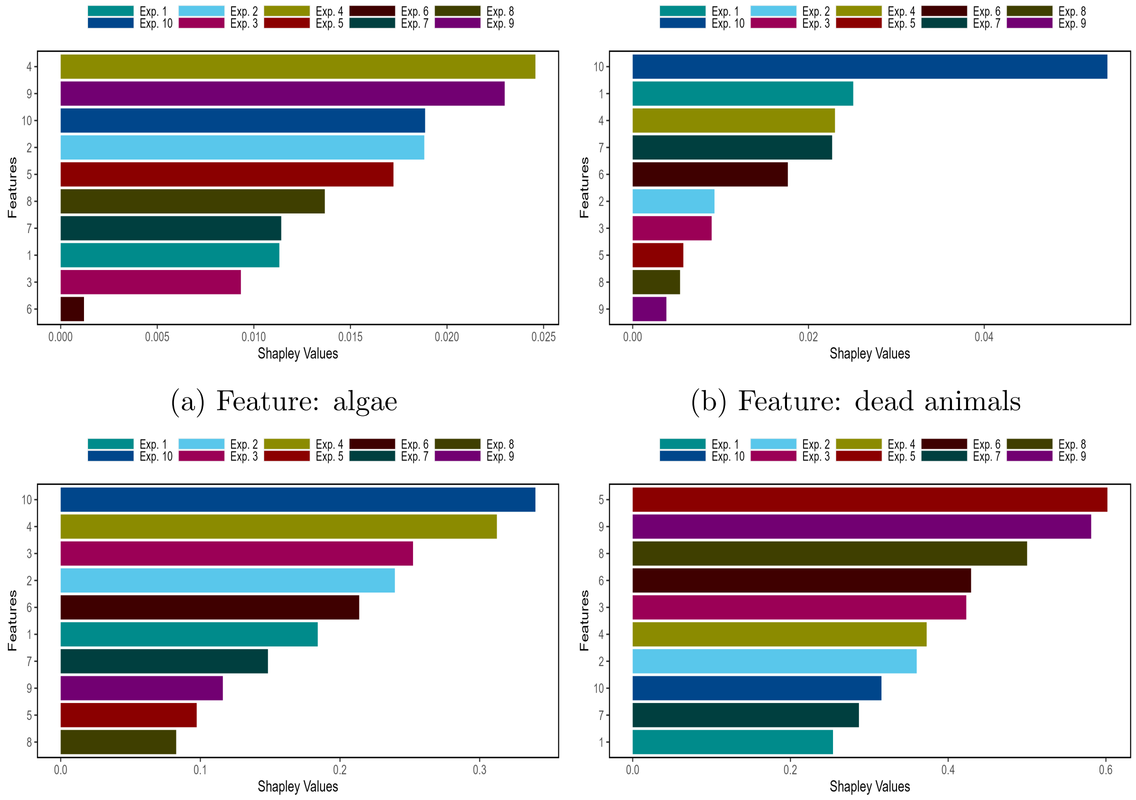
(d) Experiment 10

FIG. 33: Shapley feature importance analysis results from different experiments

describe the predator-prey scenario in my study. This data consists of time series data from ten physical experiments involving a planktonic predator-prey system, with measured population densities of the prey (unicellular algae), predator (rotifer), and predator life stage characteristics recorded over approximately 2,000 measurement days. Each experiment represented a time series that differed based on specific changes in inputs and outputs.

Figure 33 presents the importance of various features in predicting rotifer numbers. In Experiment 1, the number of eggs was the most crucial feature for predicting rotifer numbers, followed by the egg ratio. Surprisingly, the number of algae and dead animals was found to play the least important roles.

Similar results were obtained in Experiments 2, 3, 4, 6, 7, and 9, and therefore, the figures for these experiments are not included. In Experiment 5, the same feature importance pattern was observed, but the egg ratio had a slightly smaller importance value. In Experi-



(a) Feature: algae

(b) Feature: dead animals

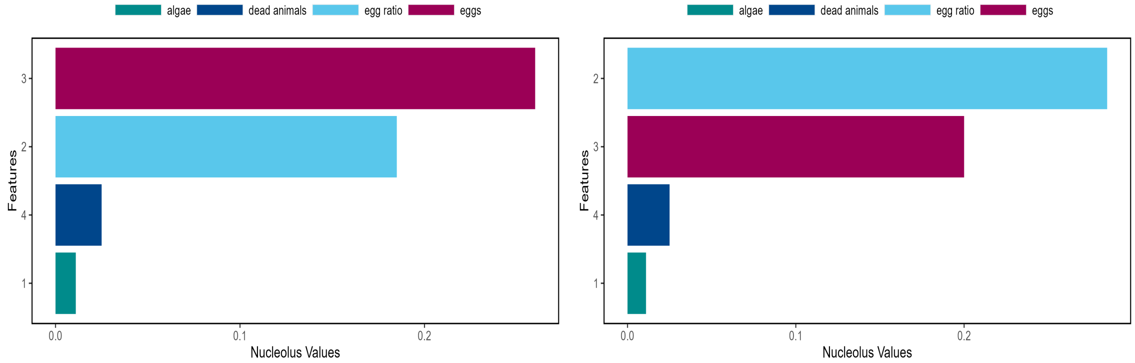
(c) Feature: egg ratio

(d) Feature: eggs

FIG. 34: Comparing Shapley feature importance values for each feature across multiple experiments.

ment 8, the feature external was identified as the third most important feature, whereas in the other experiments, it had an importance value of 0. In Experiment 8, the number of eggs was the most crucial feature for predicting rotifer numbers, followed by egg ratio and external factors. The number of algae and dead animals was again found to play the least important role. In Experiment 10, the egg ratio was found to be relatively more important than the feature eggs. This was followed by dead animals and then the number of algae.

Next, I looked at the feature importance values for selected features across different experiments that were conducted by Blasius et al., [176], to see if the same feature importance values were observed across different experiments. Notice that the features are given different feature importance values throughout different experiments. The variability in the feature importance results across different experiments suggests that the importance of different features in predicting rotifer numbers can depend on the specific conditions of each simulation. For example, factors such as the type and quantity of food provided to the rotifers, the temperature and lighting conditions, the length of the experiment, or the presence of predators can all affect the growth and reproduction of rotifers and, consequently, the importance of different features in predicting their numbers.



(a) Nucleolus: Experiment 1

(b) Nucleolus: Experiment 10

FIG. 35: Nucleolus feature importance values for Experiment 1 and Experiment 10

Subsequent to the Shapley feature importance analysis, Nucleolus future importance values were evaluated. As depicted in Figure 35, the results obtained from the NcFI method closely mirror those of the SFI values, demonstrating a high degree of similarity. Moreover, consistent with the findings from Experiment 1, the NcFI demonstrated analogous outcomes across all experiments, with Experiment 10 being the sole exception.

Next, Shapley-Shubik (SH2FI) and Banzhaf (BFI) power feature importance values were computed. Figure 36 illustrates the findings from this analysis. The data indicate that three particular features—eggs, egg ratio, and dead animals—exhibit equivalent influence on rotifer populations. In contrast, the presence of algae appears to be inconsequential for determining rotifer numbers. Additionally, it is important to note that the SH2FI and BFI indices produced identical outcomes. It is important to mention that neither SH2FI nor BFI will assign importance values when a predetermined threshold is set at a level that the features cannot attain. For instance, in a non-technical analogy, if an item is priced at $100 and there are three individuals whose combined financial contribution is $9, the importance

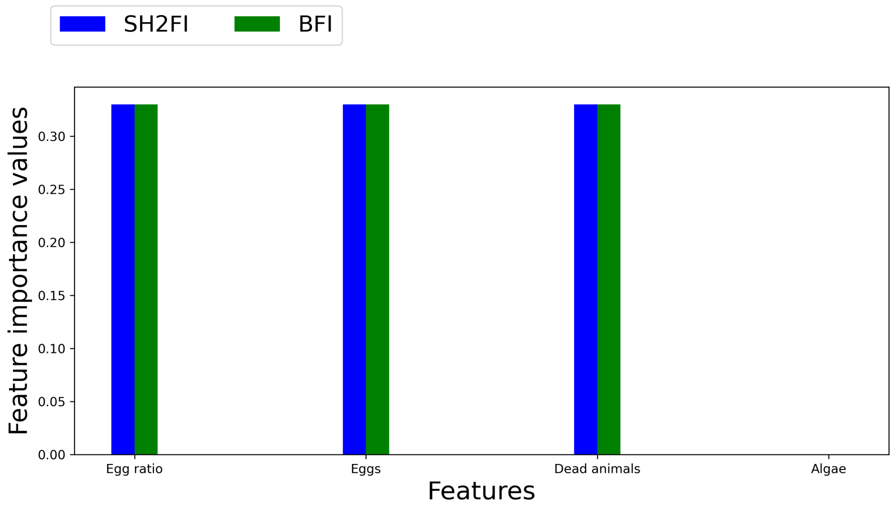


FIG. 36: Shapley-Shubik and Banzhaf power index feature importance values

value, as calculated by SH2FI and BFI, would be undefined or zero. This is because neither individually nor collectively can they meet the purchase threshold.

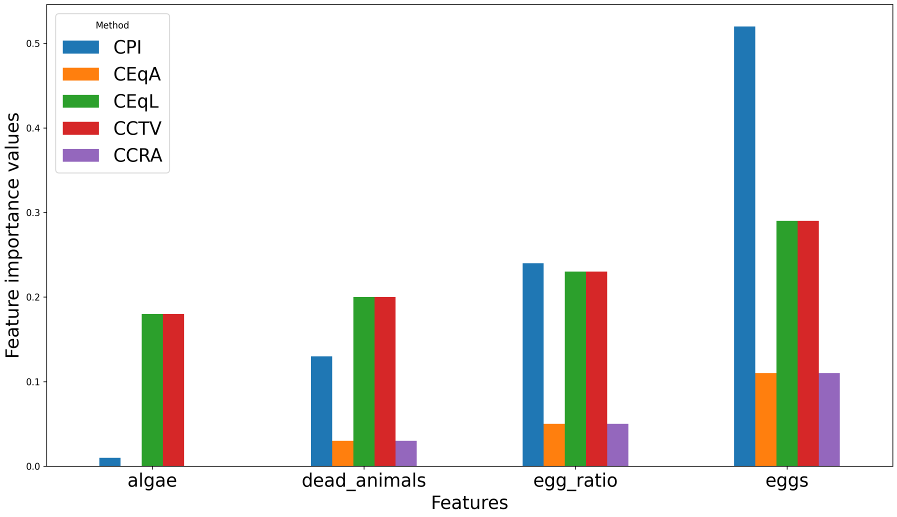


FIG. 37: Conflicting claim solution for the predator-prey experiment

Finally, feature importance methods based on conflicting claims problems were evaluated. Figure 37 suggests the same result that the reproductive status of the rotifer population, as measured by the number of eggs and the egg ratio, has the most significant impact on the population dynamics. Additionally, external factors such as the amount of algae and dead animals can also have an impact on shaping the predator-prey cycles.

Overall, the findings of this study offer preliminary insights into the predator-prey model, which could be used when designing the actual agent-based simulation models.

4.4.1 PRIME WITH WEIGHTED SHANNON ENTROPY

This section evaluates the reliability of feature importance methods by analyzing how consistently each feature importance ranking is maintained across multiple permutations. For this assessment, 60 permutations were executed. Each permutation involved randomly shuffling the values of the feature while the values of all other features were kept unchanged. After each permutation, I assessed the impact on feature importance values. In total, 240 datasets underwent this evaluation process. The feature importance method was applied to each dataset to derive the corresponding probability distributions of importance values. Tables present these probability distributions 30, 31, 32, 33, 34. In these tables, Rank 0 corresponds to the baseline ranking of feature importance values, established when no features have undergone permutation. Rank 1 is assigned following the permutation of the Eggs feature, Rank 2 after the Egg Ratio feature has been permuted, Rank 3 upon the permutation of the Dead Animals feature, and Rank 4 after altering the Algae feature through permutation.

Table 30 displays the Shapley feature importance values obtained after performing 30 permutations on the feature Eggs. Similar to the previous experiments, here as well, the feature importance values undergo significant fluctuations when that particular feature is permuted, indicating its sensitivity to the order of data. While the importance values for other features demonstrate stability across the different permutations, underscoring their consistency in contributing to the model's predictions.

TABLE 30: Predator-prey data: SFI feature importance rankings after permutations

Feature	**Rank 0**	**Rank 1**	**Rank 2**	**Rank 3**	**Rank 4**
Eggs	1	3 (90%)	1 (100%)	1 (100%)	1 (100%)
Egg ratio	2	1 (100%)	3 (90%)	2 (100%)	2 (100%)
Dead animals	3	2 (100%)	2 (100%)	4 (100%)	3 (100%)
Algae	4	4 (90%)	4 (100%)	3 (100%)	4 (100%)

TABLE 31: Predator-prey data: NcFI feature importance rankings after permutations

Feature	**Rank 0**	**Rank 1**	**Rank 2**	**Rank 3**	**Rank 4**
Eggs	1	3 (90%)	1 (100%)	1 (100%)	1 (100%)
Egg ratio	2	1 (100%)	3 (100%)	2 (100%)	2 (100%)
Dead animals	3	2 (90%)	2 (100%)	4 (100%)	3 (100%)
Algae	4	4 (100%)	4 (100%)	3 (100%)	4 (100%)

TABLE 32: Predator-prey: SH2FI and BFI importance rankings after permutations

Feature	**Rank 0**	**Rank 1**	**Rank 2**	**Rank 3**	**Rank 4**
Eggs	1	4 (100%)	1 (100%)	1 (100%)	1 (100%)
Egg ratio	1	1 (100%)	3 (100%)	2 (100%)	2 (100%)
Dead animals	1	2 (100%)	2 (100%)	4 (100%)	3 (100%)
Algae	4	3 (100%)	4 (100%)	3 (100%)	4 (100%)

TABLE 33: Predator-prey data: CPI, CEqA, CEqL, CCTV feature importance rankings after permutations

Feature	**Rank 0**	**Rank 1**	**Rank 2**	**Rank 3**	**Rank 4**
Eggs	1	3 (100%)	1 (100%)	1 (100%)	1 (100%)
Egg ratio	2	1 (100%)	3 (90%)	2 (100%)	2 (100%)
Dead animals	3	2 (95%)	2 (100%)	4 (100%)	3 (100%)
Algae	4	4 (95%)	4 (100%)	3 (100%)	4 (100%)

TABLE 34: Predator-prey data: CCRA feature importance rankings after permutations

Feature	**Rank 0**	**Rank 1**	**Rank 2**	**Rank 3**	**Rank 4**
Eggs	1	3 (75%)	1 (100%)	1 (100%)	1 (100%)
Egg ratio	2	1 (100%)	3 (100%)	2 (100%)	2 (100%)
Dead animals	3	2 (75%)	2 (100%)	4 (100%)	3 (100%)
Algae	4	4 (100%)	4 (100%)	3 (100%)	4 (100%)

In the evaluation of permutation importance, it was observed that several methods produced similar outcomes. For instance, SFI and NcFI methods yielded nearly identical rankings of feature importance across the permutations. Similar patterns of resemblance were also observed between SH2FI and BFI. Furthermore, a close resemblance was noted among feature importance methods designed to address issues related to conflicting claims, with the exception of the CCRA. CCRA exhibited the lowest probability distribution in ranking the importance values of the feature Eggs upon its permutation.

Using the data of these probability distributions, Weighted Shannon entropy values are measured. Figure 38 presents the results of the weighted Shannon entropy permutation importance evaluation for different feature importance methods applied to Predator-prey

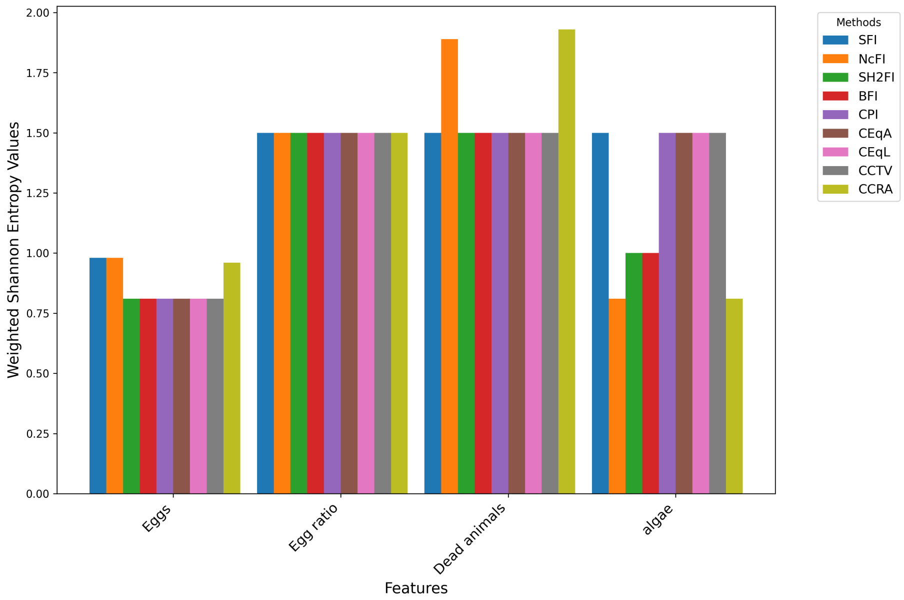


FIG. 38: Weighted Shannon entropy PRIME results based on cooperative-game theory feature importance permutation methods

data. The outcomes highlight a relatively consistent ranking of features, especially for the Egg ratio, which shows a stable importance across methods. In contrast, the ranking of Algae exhibits more variability in the permutation importance. Nonetheless, Figure 39 illustrates that, when considering the average Shannon entropy across all features and models, the results are closely aligned. A marginally lower average weighted Shannon entropy value was noted for SH2FI and BFI, with SFI registering the highest average value. Overall, this experiment recorded the lowest weighted Shannon entropy values in comparison to prior experiments.

Future work for this research is to repeat the feature importance approach but on the simulation output data. Comparing the rankings of the features (real-world data vs simulation data) might provide insight into the limitations of the simulation and/or add to the validation process of the simulation. Thus, feature importance might be able to aid the validation process beyond informing the features to focus on.

Spearman rank correlation: Finally, the Spearman rank correlation between the original ranking and permuted feature importance ranks is measured using the data observed

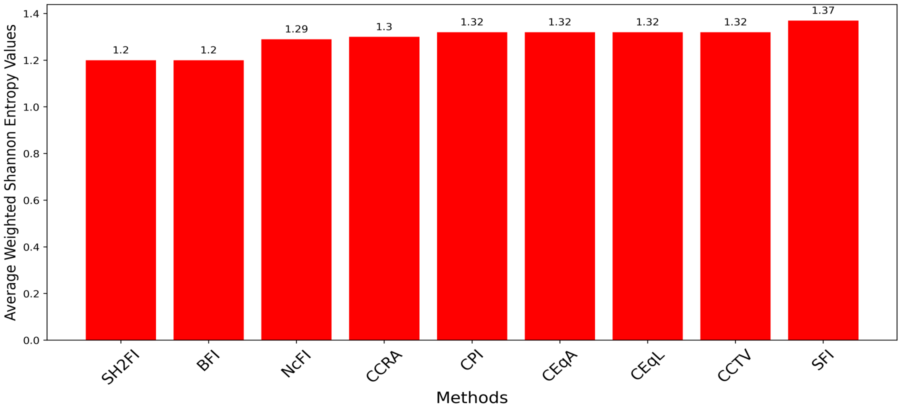


FIG. 39: Average weighted Shannon entropy PRIME results based on cooperative-game theory feature importance permutation methods

in Tables 30, 31, 32, 33, 34.

TABLE 35: Predator-prey: Spearman's rank correlation coefficient between original and permuted ranks

Permuted rankings	SFI	NcFI	Sh2FI	BFI	CPI	CEqA	CEqL	CCTV	CCRA
Rank 1	0.39	0.39	0.25	0.25	0.39	0.39	0.39	0.39	0.39
Rank 2	0.79	0.79	0.77	0.77	0.79	0.79	0.79	0.79	0.79
Rank 3	0.79	0.79	0.25	0.25	0.79	0.79	0.79	0.79	0.79
Rank 4	1	1	0.77	0.77	1	1	1	1	1

Table 35 reveals the following pattern: most permutation ranks exhibit a strong correlation with the original feature importance ranking. However, similar to the previous experiments, an exception arises in the case where the most important feature (Eggs) undergoes permutation. This specific permutation distorts the entire feature importance hierarchy, resulting in a comparatively low Spearman correlation coefficient. In other words, the original order of feature importance is not preserved after this particular permutation, emphasizing the model's reliance on feature Eggs is not replicated by other features when Eggs is permuted. This highlights the specific contribution of feature Eggs to the model's predictive performance.

In Rank 4, the permutation of the least important features (Algae) produces perfect (1) feature importance ranking correlation with the initial feature importance ranking and rankings of other feature importance methods except for SH2FI and BFI. The perfect correlation implies that the model's performance is not altered when this feature is permuted, reinforcing its lower impact on the overall predictive outcome.

Furthermore, identical correlation coefficients are observed between SFI and NcFi, SFI and SH2FI, and all the methods are based on conflicting claims solutions. This again reinforces the observations from the previous experiments that methods sharing a theoretical basis or approach to evaluating feature importance tend to produce similar results, underscoring the influence of the underlying principles on the outcomes of feature importance assessments.

4.5 CORE FEATURE IMPORTANCE

This section presents the core feature importance method, which failed to produce any results for the Seatpos, Adult, and predator-prey data, as an empty core was observed for these cases. This could be because of the features with multicollinearity that are highly substitutable, and forming a stable set with these features will be challenging. Some approaches could be employed to turn the empty core into a non-blocking coalition, such as endogenously by forming new beliefs and attitudes or exogenously from an external intervention [219].

Non-empty core: Below, an example is presented to determine the core of feature importance values where the core exists. The following is assumed: R^2 values for the features $f_1 \geq 0.2, f_2 \geq 0, f_3 \geq 0.3, f_1 + f_2 \geq 0.5, f_1 + f_3 \geq 0.8, f_2 + f_3 \geq 0.65, f_1 + f_2 + f_3 = 1$ The region of the payoff vector is represented by the smaller triangle in blue within the larger triangle in Figure 40.

Each point in Figure 40 represents an efficient payoff vector. The larger triangle shows all the possible imputations, even not satisfying the individual rationality or group rationality properties. From this set, we would like to have one imputation allocation that satisfies the individual and group rational properties. The small triangle in red is the core. In cases when the core isn't empty, it may not be unique (has multiple cores), or it may become unbounded

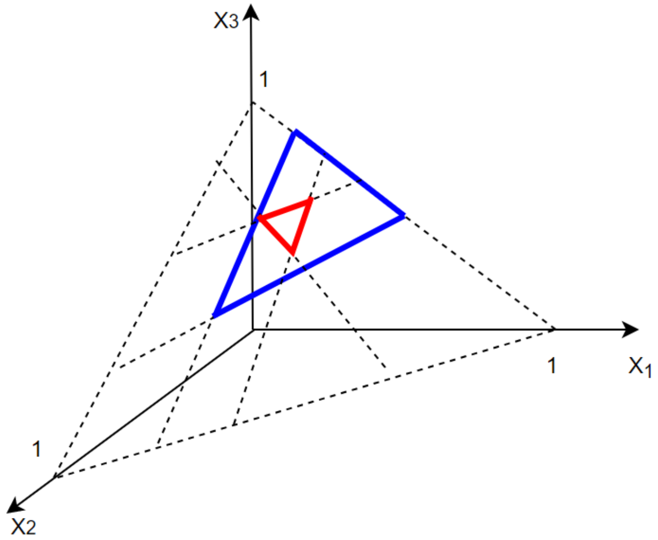


FIG. 40: Example of a core that is not empty

and have no vertices [264]. Alternative to the core is the approximate least core approach proposed by Yan and Procaccia [21]. One disadvantage of using the approximate least core approach is that it relies on random sampling. The specific allocation obtained will depend on the random subset of coalitions chosen. Therefore, the quality of the approximation depends on the number of samples taken and the randomness of the sampling. If too few samples are taken or the sampling is not random enough, the approximation may not be accurate. On the other hand, taking too many samples can be computationally expensive. In the scope of this dissertation, I do not discuss this approach nor address the empty core or multi-core problem and leave it for future studies.

4.6 RESULTS SUMMARY

This section provides a comprehensive summary of the findings from all three experiments conducted.

In summary, it becomes clear that cooperative game theory-based feature importance methods, except for the core feature importance, offer varied perspectives on the importance of features within the model. These methods illustrate how each feature contributes to the overall performance, providing explanations of their importance. The effectiveness of these methods in accurately measuring feature importance values was significantly influenced by the data, specifically whether features were correlated or independent. Tables 36, 37, 38 display the summarized rankings of feature importance values.

TABLE 36: Seatpos data: Summary of feature importance ranking generated by various methods

Model	Arm	Ht	Age	HtShoes	Seated	Leg	Thigh	Weight
SFI	6	2	8	3	4	1	6	5
NcFI	2	2	8	1	2	2	2	2
SH2FI	7	2	8	3	4	1	6	5
BFI	7	2	8	3	4	1	6	5
CPI	7	2	8	3	4	1	6	5
CEqA	1	1	1	1	1	1	1	1
CEqL	-	1	-	2	4	3	-	-
CCTV	1	1	8	1	1	1	1	1
CCRA	6	2	8	3	4	1	6	5

TABLE 37: Adult data: Summary of feature importance ranking generated by various methods

Methods	Marital status	Education	Occupation	Age	Hours per week	Sex	Native country	Work class	Race
SFI	1	2	3	4	5	6	7	8	9
NcFI	1	2	4	3	5	8	7	6	9
SH2FI	1	2	3	4	5	6	8	7	9
BFI	3	1	6	4	9	7	8	2	5
CPI	1	3	2	4	5	6	8	7	9
CEqA	1	3	2	6	4	5	8	7	9
CEqL	1	3	2	6	4	5	8	7	9
CCTV	1	3	2	6	4	5	8	7	9
CCRA	1	3	2	6	4	5	8	7	9

TABLE 38: Predator prey data: Summary of feature importance ranking generated by various methods

Methods	Eggs	Eggs ratio	Dead animals	Algae
SFI	1	2	3	4
NcFI	1	2	3	4
SH2FI	1	2	-	-
BFI	1	2	-	-
CPI	1	2	3	4
CEqA	1	2	3	-
CEqL	1	2	3	4
CCTV	1	2	3	4
CCRA	1	2	3	-

Transferable utility-based feature importance methods, such as Shapley and Nucleolus feature importance methods (SFI and NcFI), assess the contribution of each feature to the model's overall performance. Essentially, the objective of these methods is to quantify how much each feature contributes to the predictive accuracy, reduction in error, or explanatory power of the model. These methods help understand how much each variable 'adds' to the model's performance, taking into account interactions with other variables and their standalone impact. Features that contribute more to the model's performance receive a larger share of the total gains, while those with minimal impact receive smaller shares. This ensures the fair allocation of feature importance values. The SFI method has proven effective in ensuring a fair allocation of predictive power across features. It is capable of differentiating between the importance values of features, whether they are collinear or independent. The Nucleolus feature importance method was not very effective in determining the feature importance values when the features were correlated, as the features were given the same importance values across the features, this reduces the discriminatory power generated by the feature importance meth as shown in Experiment 1 (Table 36). The core feature importance method failed to generate any importance values for all the experiments that were conducted.

Voting feature importance methods are designed to assess a feature's capability to influence a specific outcome through the collective contributions of multiple features. The primary objective is to identify winning coalitions, which are combinations of features that contribute to the most effective models based on their performance capabilities. Voting feature importance methods Shapley-Shubik and Banzhaf power index feature importance methods are human-centric due to the flexibility to adjust threshold values and directly observe the changes of feature importance values.

Also, Shapley-Shubik and Banzhaf power feature importance methods are effective in addressing datasets with interdependent features. These methods exhibit a high level of discriminatory capability, distinguishing the importance of different features, in contrast to approaches like the Nucleolus method, which tends to assign equal importance to features in cases of high correlation. The term discriminatory power refers to the ability to recognize varying degrees of importance across features. The term low discriminatory power implies that all features are of equivalent importance, which might be true in certain scenarios. However, more often, features exert varying degrees of influence. When a feature importance method successfully identifies these variances, it significantly enhances the interpretability and utility of the machine learning model. Furthermore, the SH2FI and BFI methods demonstrated consistency in ranking the importance of features after the permutations, reinforcing their reliability in determining the impact of individual features.

Conflicting claims feature importance methods focus on fair allocation of the feature importance values when resources are insufficient to satisfy all claims fully. Here the resources refer to the ”capacity” of the model to incorporate and give importance weight to various features effectively. An insufficient resource means not having enough data to accurately estimate the importance of all features, limitations in computational capabilities to perform exhaustive feature importance calculations, or a model that cannot integrate all potentially informative features due to complexity or overfitting concerns. Conflicting claims problems, particularly the Constrained Equal Awards (CEqA) and Constrained Equal Losses (CEqL) feature importance methods, can sometimes fall short in efficiently determining feature importance values. This limitation stems from the fact that these methods might not fully consider the inherent predictive value or relevance of the features to the target variable, leading to allocations of importance that might be viewed as unjust in evaluating feature significance.

TABLE 39: Comparision of cooperative game theory feature importance methods developed in the scope of this dissertation: strengths and limitations

Methods	**Strengths**	**Limitation**
Shapley feature importance (SFI)	SFI method has been useful in measuring the fair distribution of the prediction ”payout” among the features, considering all possible combinations of feature presence. **Discriminatory power** is high. SFI can distinguish various feature importance values with colinear and independent data as shown in experiments 1 to 3.	**Data dependency:** SFI could be sensitive to multicollinearity. **Consistency** relatively consistent over the permutations. It is computationally intensive, especially as the number of features grows.
Nucleolus feature importance (NcFI)	NcFI method was useful in measuring the fair distribution of the prediction ”payout” among the features when the features were independent (Exp 1 and 2).	NcFI was not able to distinguish the feature importance values with **data dependencies** (Exp. 1) **Discriminatory power** is low when the data has correlations. **Consistency** in ranking the feature importance values was not stable, as evidenced by the average weighted Shannon entropy results of Experiments 1 and 2.

Continued on the next page

TABLE 39 – continued from previous page

Methods	Strengths	Limitation
Shapley-Shubik feature importance (SH2FI)	SH2FI method was useful for assessing the power dynamics and the ability to form successful coalitions. Implementation of threshold values enhances transparency and intuitiveness in examining how variations in feature contributions can significantly influence model success/failure. Exp. 1, 2, and 3 demonstrate that this method is highly human-centric due to the flexibility to adjust threshold values and directly observe the changes of feature importance values. SH2FI was useful to address data with **dependencies**, it had high **discriminatory power**, and relatively **consistent** in the feature importance ranking in all the experiments.	Similar to the SFI, this approach can be computationally intensive.

Continued on the next page

TABLE 39 – continued from previous page

Methods	**Strengths**	**Limitation**
Banzhaf power feature importance (BFI)	The BFI proved to be an effective tool for assessing feature importance in scenarios characterized by multicollinearity, as demonstrated in Exp. 1. This method exhibited **consistent** rankings of feature importance across various permutations, maintaining reliability in its evaluations as observed in Exp. 1 through 3. Similar to SH2FI, this method is highly human-centric due to the flexibility to adjust threshold values and directly observe the changes of feature importance values.	**Discriminatory power** was high for Exp. 1 and 2. However, it was low for Exp. 3. In Exp. 3, a different set of features emerged as important, diverging from the consensus typically observed. While this variation can provide unique insights, it complicates the decision-making process for the final feature selection in the model, especially when most methods converge on a similar set of features, yet the BFI highlights an entirely different subset.
CPI	CPI method was effective in measuring the feature importance values across all the experiments, including when the features were highly correlated. **Discriminatory power** was high for all the experiments.	CPI demonstrated a consistent selection of feature importance values in Exp. 2 and 3, where the features are independent. However, in Exp. 1, the consistency of the method's feature importance rankings diminished.

Continued on the next page

TABLE 39 – continued from previous page

Methods	Strengths	Limitation
CEqA	CEqA method was effective in explaining the feature importance values for experiments where data did not have any dependencies (Exp. 2 and 3).	Explanations of the feature importance values significantly struggled for Experiment 1 and generated all equal importance values. This method is sensitive to **data dependencies** and has low **discriminatory power** and **consistency** in feature importance ranking.
CEqL	CEqL method was relatively effective in identifying the feature importance values for the experiments when the data was not correlated.	This approach may not effectively manage multicollinear features, affecting the model's performance and explainability. This limits the insights that can be gained from all the futures, and it may introduce a bias towards top features. Also, with multicollinear features, the consistency of ranking was relatively unstable, and the **discriminatory power** was very low.
CCTV	CCTV method was relatively effective in identifying the feature importance values for the experiments when the data was not correlated.	Similar to CEqL, this approach may not effectively manage multicollinear features, affecting the model's performance and explainability. Also, the **discriminatory power** was very low for Exp. 1, where features had high multicollinearity.

Continued on the next page

TABLE 39 – continued from previous page

Methods	Strengths	Limitation
CCRA	CCRA method was relatively effective in identifying the feature importance values for the experiments when the data was not correlated. This method effectively manages multicollinear features, affecting the model's performance and explainability with high **discriminatory power**.	The method was not so consistent in feature importance ranking after permutations.

Overall, the SFI, SH2FI, BFI, and CPI methods demonstrated strong performance across all experiments. This involves disentangling the intertwined effects of variables, especially in cases where there are correlations or interactions between them, to identify how much each feature independently contributes to the total gains. In contrast, the NcFI method, CEqA, and CEqL faced challenges in pinpointing the individual contributions and distinct impacts of each feature under conditions of high correlation among features.

4.7 RESEARCH SUMMARY

The summary of this dissertation is presented here.

The goal of this study was to develop cooperative game theory (CGT) based explainable artificial intelligence (XAI) methods and address the research questions outlined in Chapter 1.

Chapter 2 delves into the foundational concepts of CGT and XAI, with a particular emphasis on the feature importance method utilized in XAI to determine the relevance of features within a machine learning model. This chapter aims to provide a comprehensive background, setting the stage for understanding how CGT principles can be applied to enhance interpretability in machine learning through attribution (measure) of feature importance values.

Chapter 3 presents the CGT-based feature importance methods alongside their underlying algorithms. This chapter focuses on how CGT principles are applied to identify and evaluate the relevance of individual features in enhancing the interpretability of machine learning models. Further, the chapter presents a weighted Shannon entropy-based permutation relative importance evaluation (PRIME) metric that is used to evaluate the effectiveness of feature importance methods developed in this dissertation. This discussion is anchored around two pivotal elements of the metric: weighted Shannon entropy and permutation tests, offering a detailed examination of their roles in evaluating feature importance methods.

Chapter 4 provides the study results and demonstrates that the methodology described in Chapter 3. Three different experiments are conducted with different data types and models. Experiment 1 evaluates feature importance methods utilizing linear regression models, particularly focusing on data with dependency among features. Experiment 2 investigates the application of feature importance methods within the framework of logistic regression models, where features are independent. Lastly, Experiment 3 revisits the linear regression model to explore input data that could potentially inform the design of agent-based simulation models, emphasizing practical applicability in more complex modeling scenarios. This structured approach allows for a nuanced examination of feature importance methods across different model types and data conditions, providing insights into their effectiveness and adaptability. The findings of this study reveal that a number of feature importance methodologies grounded in cooperative game theory—specifically the Shapley-Shubik index, the Banzhaf power index, and approaches based on conflicting claims—prove effective for quantifying the significance of features. These methods stand out for their user-centered design, offering the flexibility to modify threshold values and directly monitor the impact on feature importance evaluations. This adaptability enhances their applicability in real-world scenarios, allowing for more tailored analyses in understanding the contributions of individual features within machine learning models.

CHAPTER 5

CONCLUSION

This chapter concludes the work presented in this dissertation and outlines the limitations and future directions for cooperative game theory-based explainable artificial intelligence methods.

5.1 CONCLUSION

Machine learning (ML) models are used to make highly crucial decisions, varying from medical diagnosis to cyber-physical systems analysis. Understanding the decision-making process of these models will provide more knowledge and confidence about our conclusions. However, when we are dealing with some data that is twisted, such as having some multicollinearity issue, or when we are using black-box models, the explanation of the model's output may get biased and altered. Having a trustworthy and transparent model that we can rely on is crucial. Explainable artificial intelligence (XAI) has been useful in addressing these issues. XAI provides different methods to better describe the inner workings of the models as well as the model outcomes.

This dissertation develops XAI methods using cooperative game theory-based solutions with linear regression and logistic regression models. The regression model is discussed because these models become dubious when highly correlated features are present. The logistic regression model is considered to show that these methods can also be appropriate for classification tasks. Further, these methods for assessing feature importance are employed to explore the changes in input data that could enhance the predator-prey model within agent-based simulations. This approach could lead to the development of more accurate and reliable ABM simulation models.

Considering different cooperative game theory-based feature importance methods can help uncover new insights about the predictions. The choice of techniques depends on the data specifications, as some techniques may offer clear explanations for certain data while being less effective for others (e.g., Nucleolus). Incorporating approaches such as Shapley-Shubik and Banzhaf's power index can make the model more explainable and transparent, as they allow for experimentation with different threshold values to observe how feature importance values vary with adjustments. The proportional rule, constrained equal awards and losses and random arrival-based feature importance methods appear to be more consistent even in the presence of data dependencies, making them valuable approaches for measuring feature importance values.

Evaluating different feature importance methods requires considering their performance compared to other methods, their applicability to various datasets, and their ability to handle specific challenges such as multicollinearity among features. This study presents a permutation relative importance evaluation (PRIME) metric using a weighted Shannon entropy to measure the uncertainty and overall consistency associated with feature importance rankings. Higher weighted Shannon entropy values were observed for the Nucleolus feature importance method, highlighting the sensitivity of this approach in evaluating feature importance. In the dataset where the features were highly correlated, some of the methods were not very efficient, such as constrained equal awards, constrained equal losses, and conflicting claims Talmud valuation, while they were useful for the dataset where there was no dependency between the features.

5.2 LIMITATIONS AND FUTURE WORK

Employing cooperative game theory in explainable artificial intelligence (XAI) offers new insights into machine learning models and their predictions. However, these methods are contingent upon certain assumptions, and deviations from these assumptions could pose challenges that need to be addressed. The explanations generated by these methods can differ significantly from one another. The task of selecting the most appropriate method and the specific type of explanation it yields remains an open area for further investigation.

Also, the selection of the method depends on the problem of interest and the data type. For example, logistic regression-based feature importance methods performed well when the experiment was conducted, however, its performance indicator of McFadden's R-squared has limitations compared to other measures. One key limitation is that McFadden's R-squared can be relatively small even when the model has strong predictors [265]. This is because it measures the improvement of the model over a null model rather than the variance explained by the model. For example, even with strong effects on the probability of the outcome, McFadden's R-squared may not approach near 1, indicating that it might be challenging to achieve high values for this measure in practice, reflecting the inherent difficulty in predicting a binary event with certainty. In the future, other measures like Cox and Snell's R-squared and Tjur's R-squared could be explored [151].

Additionally, assessing the effectiveness of feature-importance methods continues to pose significant challenges. In this work, I took a step toward a quantitative evaluation of methods of future importance using permutation tests and the Shannon entropy approach. This work is only a starting point; one can also develop other measures of performance using the Weighted Shannon Entropy-based Permutation Importance Evaluation (PRIME) metric. Developing ways to quantitatively evaluate feature importance methods could aid in selecting the most effective method for identifying essential features. The Weighted Shannon Entropy-based PRIME metric compares various feature importance methods by evaluating feature importance rankings on an individual basis. One of the limitations associated with PRIME is that it employs a majority ranking system for the feature importance distributions. This means the rank of a feature is determined by the most frequent ranking positions across different permutation importance evaluations [263]. By focusing solely on the majority ranking, this metric might disregard important insights from models that have identified different feature importance rankings, especially when the majority consensus is not overwhelming. Incorporating the feature importance rankings from minority rankings could ensure that less common but potentially insightful rankings are not overlooked. Note that close rankings, where the difference in agreement levels is minimal (e.g., 55% vs. 45%), are relatively rare. In most cases, the majority ranking method yields clear and decisive outcomes, with feature rankings often selected with high levels of confidence. This is an essential aspect of the method's reliability, as it implies that in the vast majority of cases, there is a significant consensus among feature importance methods regarding the importance of features. Also, tied rankings can arise when two or more features exhibit identical levels of importance across various models or evaluations. This phenomenon has been noted in scenarios where features show a high correlation and in certain methodologies, including Nucleolus Feature Importance (NcFI). However, this could be due to the objective of NcFI, which is to minimize 'unhappiness' among models that receive the least benefit [28]. The term 'model unhappiness' refers to the gap between the highest outcomes and the actual results received, indicating underperformance. Addressing these limitations could further improve the reliability of evaluations of feature importance methods.

In the future, this metric could accommodate various ranking methods. Also, normalizing the weighted probabilities may lead to some information loss, particularly if there are extreme differences in the frequencies of rankings. This normalization may not fully capture the nuances of the feature importance values, potentially resulting in oversimplified entropy calculations. Also, some of these methods may be sensitive to outliers in the feature importance ranking, potentially skewing the entropy calculations. Robustness measures may need to be implemented to mitigate the impact of outliers on the results.

Finally, these feature importance methods developed in this dissertation could be applied to more complex models such as computer vision, large-scale language models, and non-linear problems. For computer vision, this could mean identifying which pixels or patterns are most influential in image recognition tasks, leading to better model interpretability and enhanced training strategies. In the realm of large-scale language models, feature importance can reveal which words or phrases carry the most weight in determining the context or sentiment of text, facilitating the refinement of models for greater accuracy and efficiency. These feature importance methods could open avenues for optimizing model performance and reliability.

BIBLIOGRAPHY

[1] I. Basu and S. Maji, "Multicollinearity correction and combined feature effect in shapley values," in *Australasian Joint Conference on Artificial Intelligence*, pp. 79–90, Springer, 2022.

[2] B. Mahesh, "Machine learning algorithms-a review," *International Journal of Science and Research (IJSR).[Internet]*, vol. 9, no. 1, pp. 381–386, 2020.

[3] D. Janzing, L. Minorics, and P. Blöbaum, "Feature relevance quantification in explainable ai: A causal problem," in *International Conference on artificial intelligence and statistics*, pp. 2907–2916, PMLR, 2020.

[4] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, "A survey of methods for explaining black box models," *ACM computing surveys (CSUR)*, vol. 51, no. 5, pp. 1–42, 2018.

[5] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, *et al.*, "Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai," *Information fusion*, vol. 58, pp. 82–115, 2020.

[6] W. Samek, G. Montavon, A. Vedaldi, L. K. Hansen, and K.-R. Müller, *Explainable AI: interpreting, explaining and visualizing deep learning*, vol. 11700. Springer Nature, 2019.

[7] L. Wells and T. Bednarz, "Explainable ai and reinforcement learning—a systematic review of current approaches and trends," *Frontiers in artificial intelligence*, vol. 4, p. 550030, 2021.

[8] J. Ranstam and J. Cook, "Lasso regression," *Journal of British Surgery*, vol. 105, no. 10, pp. 1348–1348, 2018.

[9] G. C. McDonald, "Ridge regression," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 1, no. 1, pp. 93–100, 2009.

[10] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and intelligent laboratory systems*, vol. 2, no. 1-3, pp. 37–52, 1987.

[11] S. Prasad and L. M. Bruce, "Limitations of principal components analysis for hyperspectral target recognition," *IEEE Geoscience and Remote Sensing Letters*, vol. 5, no. 4, pp. 625–629, 2008.

[12] V. Fonti and E. Belitser, "Feature selection using lasso," *VU Amsterdam research paper in business analytics*, vol. 30, pp. 1–25, 2017.

[13] M. T. Ribeiro, S. Singh, and C. Guestrin, "" why should i trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

[14] I. Covert, S. M. Lundberg, and S.-I. Lee, "Understanding global feature contributions with additive importance measures," *Advances in Neural Information Processing Systems*, vol. 33, pp. 17212–17223, 2020.

[15] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Advances in neural information processing systems*, vol. 30, 2017.

[16] C. Molnar, *Interpretable machine learning.* Lulu. com, 2020.

[17] Y. Kwon and J. Y. Zou, "Weightedshap: analyzing and improving shapley based feature attributions," *Advances in Neural Information Processing Systems*, vol. 35, pp. 34363–34376, 2022.

[18] C. Frye, C. Rowat, and I. Feige, "Asymmetric shapley values: incorporating causal knowledge into model-agnostic explainability," *Advances in Neural Information Processing Systems*, vol. 33, pp. 1229–1239, 2020.

[19] M. Sundararajan, A. Taly, and Q. Yan, “Axiomatic attribution for deep networks,” in *International conference on machine learning*, pp. 3319–3328, PMLR, 2017.

[20] Y. Zhou, S. Booth, M. T. Ribeiro, and J. Shah, “Do feature attribution methods correctly attribute features?,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, pp. 9623–9633, 2022.

[21] T. Yan and A. D. Procaccia, “If you like shapley then you’ll love the core,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 5751–5759, 2021.

[22] A. Adadi and M. Berrada, “Peeking inside the black-box: A survey on explainable artificial intelligence (xai),” *IEEE ACCESS*, vol. 6, pp. 52138–52160, 2018.

[23] F. Yan, S. Wen, S. Nepal, C. Paris, and Y. Xiang, “Explainable machine learning in cybersecurity: A survey,” *International Journal of Intelligent Systems*, 2022.

[24] L. Shapley, “A value for n-person games,” *Contributions to the Theory of Games*, vol. 28, no. 2, pp. 307–317, 1953.

[25] D. Fryer, I. Strümke, and H. Nguyen, “Shapley values for feature selection: The good, the bad, and the axioms,” *IEEE Access*, vol. 9, pp. 144352–144360, 2021.

[26] I. E. Kumar, S. Venkatasubramanian, C. Scheidegger, and S. Friedler, “Problems with shapley-value-based explanations as feature importance measures,” in *International Conference on Machine Learning*, pp. 5491–5500, PMLR, 2020.

[27] G. Chalkiadakis, E. Elkind, and M. Wooldridge, “Computational aspects of cooperative game theory,” *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 5, no. 6, pp. 1–168, 2011.

[28] M. Maschler, “The bargaining set, kernel, and nucleolus,” *Handbook of game theory with economic applications*, vol. 1, pp. 591–667, 1992.

[29] F. Turnovec, J. W. Mercik, M. Mazurkiewicz, *et al.*, "Power indices: Shapley-shubik or penrose-banzhaf," *Operational Research and Systems*, pp. 121–127, 2004.

[30] P. Dubey and L. S. Shapley, "Mathematical properties of the banzhaf power index," *Mathematics of Operations Research*, vol. 4, no. 2, pp. 99–131, 1979.

[31] D. Wang, Q. Yang, A. Abdul, and B. Y. Lim, "Designing theory-driven user-centric explainable ai," in *Proceedings of the 2019 CHI conference on human factors in computing systems*, pp. 1–15, 2019.

[32] Q. V. Liao and K. R. Varshney, "Human-centered explainable ai (xai): From algorithms to user experiences," *arXiv preprint arXiv:2110.10790*, 2021.

[33] O. O. Aalen, "A linear regression model for the analysis of life times," *Statistics in medicine*, vol. 8, no. 8, pp. 907–925, 1989.

[34] S. Menard, *Applied logistic regression analysis*. No. 106, Sage, 2002.

[35] P. C. Sen, M. Hajra, and M. Ghosh, "Supervised classification algorithms in machine learning: A survey and review," in *Emerging Technology in Modelling and Graphics: Proceedings of IEM Graph 2018*, pp. 99–111, Springer, 2020.

[36] C. M. Macal and M. J. North, "Tutorial on agent-based modeling and simulation," in *Proceedings of the 2005 Winter Simulation Conference* (M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines, eds.), (Piscataway, New Jersey), pp. 14–pp, Institute of Electrical and Electronics Engineers, Inc., 2005.

[37] S. Lipovetsky and M. Conklin, "Analysis of regression in game theory approach," *Applied Stochastic Models in Business and Industry*, vol. 17, no. 4, pp. 319–330, 2001.

[38] C. M. Macal and M. J. North, "Agent-based modeling and simulation," in *Proceedings of the 2009 Winter Simulation Conference* (M. D. Rossetti, R. R. Hill, B. Johansson,

A. Dunkin, and R. G. Ingalls, eds.), (Piscataway, New Jersey), pp. 86–98, Institute of Electrical and Electronics Engineers, Inc., 2009.

[39] J. C. Thiele, W. Kurth, and V. Grimm, "Facilitating parameter estimation and sensitivity analysis of agent-based models: A cookbook using netlogo and r," *Journal of Artificial Societies and Social Simulation*, vol. 17, no. 3, p. 11, 2014.

[40] G. Ten Broeke, G. Van Voorn, and A. Ligtenberg, "Which sensitivity analysis method should i use for my agent-based model?," *Journal of Artificial Societies and Social Simulation*, vol. 19, no. 1, p. 5, 2016.

[41] K. Shailaja, B. Seetharamulu, and M. Jabbar, "Machine learning in healthcare: A review," in *2018 Second international conference on electronics, communication and aerospace technology (ICECA)*, pp. 910–914, IEEE, 2018.

[42] F. García-Peñalvo, J. Cruz-Benito, M. Martín-González, A. Vázquez-Ingelmo, J. C. Sánchez-Prieto, and R. Therón, "Proposing a machine learning approach to analyze and predict employment and its factors," 2018.

[43] S. Lee, Y. Kim, H. Kahng, S.-K. Lee, S. Chung, T. Cheong, K. Shin, J. Park, and S. B. Kim, "Intelligent traffic control for autonomous vehicle systems based on machine learning," *Expert Systems with Applications*, vol. 144, p. 113074, 2020.

[44] C. J. Lynch, R. Gore, A. J. Collins, T. S. Cotter, G. Grigoryan, and J. F. Leathrum, "Increased need for data analytics education in support of verification and validation," in *Proceedings of the 2021 Winter Simulation Conference* (S. Kim, B. Feng, K. Smith, S. Masoud, Z. Zheng, C. Szabo, and M. Loper, eds.), (Piscataway, New Jersey), pp. 1–12, Institute of Electrical and Electronics Engineers, Inc., 2021.

[45] R. Berk, *Criminal justice forecasts of risk: A machine learning approach.* Springer Science & Business Media, 2012.

[46] N. Kordzadeh and M. Ghasemaghaei, "Algorithmic bias: review, synthesis, and future research directions," *European Journal of Information Systems*, vol. 31, no. 3, pp. 388–409, 2022.

[47] T. Panch, H. Mattie, and R. Atun, "Artificial intelligence and algorithmic bias: implications for health systems," *Journal of global health*, vol. 9, no. 2, 2019.

[48] M. O. Riedl, "Human-centered artificial intelligence and machine learning," *Human behavior and emerging technologies*, vol. 1, no. 1, pp. 33–36, 2019.

[49] H. Lakkaraju and O. Bastani, "" how do i fool you?" manipulating user trust via misleading black box explanations," in *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pp. 79–85, 2020.

[50] G. Grigoryan, "Explainable artificial intelligence: Requirements for explainability," in *Proceedings of the 2022 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation*, pp. 27–28, 2022.

[51] O. Loyola-Gonzalez, "Black-box vs. white-box: Understanding their advantages and weaknesses from a practical point of view," *IEEE Access*, vol. 7, pp. 154096–154113, 2019.

[52] D. Doran, S. Schulz, and T. R. Besold, "What does explainable ai really mean? a new conceptualization of perspectives," *arXiv preprint arXiv:1710.00794*, 2017.

[53] I. El Naqa and M. J. Murphy, "What is machine learning?," in *machine learning in radiation oncology*, pp. 3–11, Springer, 2015.

[54] M. A. Audette, T. Rashid, S. Ghosh, N. Patel, and S. Sultana, "Towards an anatomical modeling pipeline for simulation and accurate navigation for brain and spine surgery," in *Proceedings of the Summer Simulation Multi-Conference*, pp. 1–12, 2017.

[55] S. Ghosh, Y. Chen, and W. Dou, "Railroad safety: A systematic analysis of twitter data," *Case Studies on Transport Policy*, vol. 15, p. 101154, 2024.

[56] C. O'Neil, "The era of blind faith in big data must end," *TED2017. Retrieved from: https://archive. org/details/CathyONeil_2017*, 2017.

[57] C. Molnar, "A guide for making black box models explainable," *URL: https://christophm. github. io/interpretable-ml-book*, vol. 2, no. 3, 2018.

[58] R. Roscher, B. Bohn, M. F. Duarte, and J. Garcke, "Explainable machine learning for scientific insights and discoveries," *Ieee Access*, vol. 8, pp. 42200–42216, 2020.

[59] V. Belle and I. Papantonis, "Principles and practice of explainable machine learning," *Frontiers in big Data*, p. 39, 2021.

[60] S. Gregor and I. Benbasat, "Explanations from intelligent systems: Theoretical foundations and implications for practice," *MIS quarterly*, pp. 497–530, 1999.

[61] R. C. Schank, "Explanation: A first pass," *Experience, memory, and reasoning*, pp. 139–165, 1986.

[62] D. Gunning, "Explainable artificial intelligence (xai)(2017)," *Seen on*, vol. 1, 2017.

[63] A. S. Ross, M. C. Hughes, and F. Doshi-Velez, "Right for the right reasons: Training differentiable models by constraining their explanations," *arXiv preprint arXiv:1703.03717*, 2017.

[64] A. Rosenfeld and A. Richardson, "Explainability in human–agent systems," *Autonomous Agents and Multi-Agent Systems*, vol. 33, no. 6, pp. 673–705, 2019.

[65] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal, "Explaining explanations: An overview of interpretability of machine learning," in *2018 IEEE 5th International Conference on data science and advanced analytics (DSAA)*, pp. 80–89, IEEE, 2018.

[66] R. Confalonieri, L. Coba, B. Wagner, and T. R. Besold, "A historical perspective of explainable artificial intelligence," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 11, no. 1, p. e1391, 2021.

[67] M. Langer, D. Oster, T. Speith, H. Hermanns, L. Kästner, E. Schmidt, A. Sesing, and K. Baum, "What do we want from explainable artificial intelligence (xai)?–a stakeholder perspective on xai and a conceptual model guiding interdisciplinary xai research," *Artificial Intelligence*, vol. 296, p. 103473, 2021.

[68] D. Shin, "The effects of explainability and causability on perception, trust, and acceptance: Implications for explainable ai," *International Journal of Human-Computer Studies*, vol. 146, p. 102551, 2021.

[69] A. Kelly-Lyth, "Challenging biased hiring algorithms," *Oxford Journal of Legal Studies*, vol. 41, no. 4, pp. 899–928, 2021.

[70] G. Grigoryan, L. Robaldo, A. Pinto, and A. J. Collins, "Exploring the explainability and legal implications of regression models in transportation domain," in *Jurisinformatics* (JURISIN), Workshop publication, 2023.

[71] S. O'Sullivan, N. Nevejans, C. Allen, A. Blyth, S. Leonard, U. Pagallo, K. Holzinger, A. Holzinger, M. I. Sajid, and H. Ashrafian, "Legal, regulatory, and ethical frameworks for development of standards in artificial intelligence (ai) and autonomous robotic surgery," *The international journal of medical robotics and computer assisted surgery*, vol. 15, no. 1, p. e1968, 2019.

[72] G. König, C. Molnar, B. Bischl, and M. Grosse-Wentrup, "Relative feature importance," in *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 9318–9325, IEEE, 2021.

[73] A. Bell, I. Solano-Kamaiko, O. Nov, and J. Stoyanovich, "It's just not that simple: an empirical study of the accuracy-explainability trade-off in machine learning for public

policy," in *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, pp. 248–266, 2022.

[74] B. Letham, C. Rudin, T. H. McCormick, and D. Madigan, "Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model," *The Annals of Applied Statistics*, vol. 9, no. 3, pp. 1350–1371, 2015.

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

[76] U. Johansson, R. König, and L. Niklasson, "The truth is in there-rule extraction from opaque models using genetic programming.," in *FLAIRS Conference*, pp. 658–663, Miami Beach, FL, 2004.

[77] P. Sadowski, J. Collado, D. Whiteson, and P. Baldi, "Deep learning, dark knowledge, and dark matter," in *NIPS 2014 Workshop on High-energy Physics and Machine Learning*, pp. 81–87, PMLR, 2015.

[78] Y. Zhang and B. Wallace, "A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification," *arXiv preprint arXiv:1510.03820*, 2015.

[79] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek, "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation," *PloS one*, vol. 10, no. 7, p. e0130140, 2015.

[80] C. Aldrich, "Process variable importance analysis by use of random forests in a shapley regression framework," *Minerals*, vol. 10, no. 5, p. 420, 2020.

[81] L. Breiman, "Random forests," *Machine learning*, vol. 45, pp. 5–32, 2001.

[82] A. Altmann, L. Toloşi, O. Sander, and T. Lengauer, "Permutation importance: a

corrected feature importance measure," *Bioinformatics*, vol. 26, no. 10, pp. 1340–1347, 2010.

[83] L. S. Shapley *et al.*, "A value for n-person games," 1953.

[84] I. Covert and S.-I. Lee, "Improving kernelshap: Practical shapley value estimation using linear regression," in *International Conference on Artificial Intelligence and Statistics*, pp. 3457–3465, PMLR, 2021.

[85] G. Grigoryan and A. J. Collins, "Game theory for systems engineering: A survey," *International Journal of System of Systems Engineering*, vol. 11, no. 2, pp. 121–158, 2021.

[86] J. Von Neumann and O. Morgenstern, "Theory of games and economic behavior, 2nd rev," 1947.

[87] D. Fudenberg and J. Tirole, *Game Theory.* MIT press, 1991.

[88] G. Owen, *Game Theory.* Emerald Group Publishing, 2013.

[89] Y. Shoham and K. Leyton-Brown, *Multiagent systems: Algorithmic, game-theoretic, and logical foundations.* Cambridge University Press, 2008.

[90] R. B. Myerson, *Game theory: analysis of conflict.* Harvard university press, 1997.

[91] R. J. Aumann and S. Hart, *Handbook of game theory with economic applications*, vol. 2. Elsevier, 1992.

[92] K. W. Hipel and A. Obeidi, "Trade versus the environment: Strategic settlement from a systems engineering perspective," *Systems Engineering*, vol. 8, no. 3, pp. 211–233, 2005.

[93] J. C. Harsanyi, “Paradoxes of rationality: Theory of metagames and political behavior. by nigel howard.(cambridge, mass.: Mit press, 1971. pp. 248. 12.95.),″AmericanPoliticalScienceReview, vol. 67, no. 2, pp. 599 − −600, 1973.

[94] J. Bryant, “The plot thickens: understanding interaction through the metaphor of drama,” *Omega*, vol. 25, no. 3, pp. 255–266, 1997.

[95] L. Raiffa and R. D. Luce, *Games and decisions.* Wiley New York, 1957.

[96] T. Başar and G. J. Olsder, *Dynamic noncooperative game theory.* SIAM, 1998.

[97] R. Selten and R. S. Bielefeld, *Reexamination of the perfectness concept for equilibrium points in extensive games.* Springer, 1988.

[98] R. J. Aumann, “Correlated equilibrium as an expression of bayesian rationality,” *Econometrica: Journal of the Econometric Society*, pp. 1–18, 1987.

[99] J. Nash, “Non-cooperative games,” *Annals of mathematics*, pp. 286–295, 1951.

[100] A. J. Collins, T. Thomas, and G. Grigoryan, “Monte carlo simulation of hedonic games,” in *MODSIM World 2019 Conference, Norfolk, VA, USA*, 2019.

[101] A. Collins, W. Jayanetti, G. Grigoryan, and D. Chatfield, “Using a machine learning approach to advance agent-based simulation in a cooperative game theory context,” in *IISE Annual Conference and Expo*, IISE, 2023.

[102] R. Kamalapurkar, P. Walters, and W. E. Dixon, “Model-based reinforcement learning for approximate optimal regulation,” *Automatica*, vol. 64, pp. 94–104, 2016.

[103] V. Vinoba and S. Sridevi, “A bargaining cooperative and non-cooperative game theory in wireless sensor networks,” *International Journal of Mathematics And its Applications*, vol. 5, no. 4-C, pp. 321–330, 2017.

[104] W. Saad, Z. Han, M. Debbah, A. Hjorungnes, and T. Basar, “Coalitional game theory for communication networks,” *Ieee signal processing magazine*, vol. 26, no. 5, pp. 77–97, 2009.

[105] S. Airiau, “Cooperative games and multiagent systems,” *The Knowledge Engineering Review*, vol. 28, no. 4, pp. 381–424, 2013.

[106] S. Cano-Berlanga, J.-M. Giménez-Gómez, and C. Vilella, “Enjoying cooperative games: The r package gametheory,” *Applied Mathematics and Computation*, vol. 305, pp. 381–393, 2017.

[107] S. Banerjee, H. Konishi, and T. Sönmez, “Core in a simple coalition formation game,” *Social Choice and Welfare*, vol. 18, pp. 135–153, 2001.

[108] A. Bogomolnaia and M. O. Jackson, “The stability of hedonic coalition structures,” *Games and Economic Behavior*, vol. 38, no. 2, pp. 201–230, 2002.

[109] C. Ballester, “Np-completeness in hedonic games,” *Games and Economic Behavior*, vol. 49, no. 1, pp. 1–30, 2004.

[110] D. B. Gillies, “Solutions to general non-zero-sum games,” *Contributions to the Theory of Games*, vol. 4, pp. 47–85, 1959.

[111] L. C. Thomas, *Games, theory and applications.* Courier Corporation, 2012.

[112] H. P. Young, “Monotonic solutions of cooperative games,” *International Journal of Game Theory*, vol. 14, no. 2, pp. 65–72, 1985.

[113] M. Karakaya, “Hedonic coalition formation games: A new stability notion,” *Mathematical Social Sciences*, vol. 61, no. 3, pp. 157–165, 2011.

[114] M. J. Osborne and A. Rubinstein, *A course in game theory.* MIT press, 1994.

[115] R. J. Aumann and M. Maschler, “Game theoretic analysis of a bankruptcy problem from the talmud,” *Journal of economic theory*, vol. 36, no. 2, pp. 195–213, 1985.

[116] L. Shapley, "Quota solutions op n-person games1," *Edited by Emil Artin and Marston Morse*, p. 343, 1953.

[117] V. Mazalov, *Mathematical game theory and applications.* John Wiley & Sons, 2014.

[118] L. S. Shapley, "On balanced sets and cores," tech. rep., RAND CORP SANTA MONICA CALIF, 1965.

[119] O. N. Bondareva, "Some applications of linear programming methods to the theory of cooperative games," *Problemy kibernetiki*, vol. 10, no. 119, p. 139, 1963.

[120] D. Schmeidler, "The nucleolus of a characteristic function game," *SIAM Journal on applied mathematics*, vol. 17, no. 6, pp. 1163–1170, 1969.

[121] M. Zuckerman, P. Faliszewski, Y. Bachrach, and E. Elkind, "Manipulating the quota in weighted voting games," *Artificial Intelligence*, vol. 180, pp. 1–19, 2012.

[122] L. S. Shapley and M. Shubik, "A method for evaluating the distribution of power in a committee system," *American political science review*, vol. 48, no. 3, pp. 787–792, 1954.

[123] L. S. Penrose, "The elementary statistics of majority voting," *Journal of the Royal Statistical Society*, vol. 109, no. 1, pp. 53–57, 1946.

[124] J. F. Banzhaf III, "Weighted voting doesn't work: A mathematical analysis," *Rutgers L. Rev.*, vol. 19, p. 317, 1964.

[125] B. O'Neill, "A problem of rights arbitration from the talmud," *Mathematical social sciences*, vol. 2, no. 4, pp. 345–371, 1982.

[126] N. Dagan and O. Volij, "The bankruptcy problem: a cooperative bargaining approach," *Mathematical Social Sciences*, vol. 26, no. 3, pp. 287–297, 1993.

[127] W. Thomson, "Axiomatic and game-theoretic analysis of bankruptcy and taxation problems: a survey," *Mathematical social sciences*, vol. 45, no. 3, pp. 249–297, 2003.

[128] J.-M. Giménez-Gómez, J. Teixidó-Figueras, and C. Vilella, "The global carbon budget: a conflicting claims problem," *Climatic change*, vol. 136, pp. 693–703, 2016.

[129] M. Á. M. Calvo, I. N. Lugilde, C. Q. Sandomingo, and E. S. Rodríguez, "An operational toolbox for solving conflicting claims problems," *Decision Analytics Journal*, vol. 6, p. 100160, 2023.

[130] M. C. Gallastegui, E. Inarra, and R. Prellezo, "Bankruptcy of fishing resources: the northern european anglerfish fishery," *Marine Resource Economics*, vol. 17, no. 4, pp. 291–307, 2002.

[131] D. G. McVitie and L. B. Wilson, "The stable marriage problem," *Communications of the ACM*, vol. 14, no. 7, pp. 486–490, 1971.

[132] R. W. Irving, "An efficient algorithm for the "stable roommates" problem," *Journal of Algorithms*, vol. 6, no. 4, pp. 577–595, 1985.

[133] V. P. Crawford, "The flexible-salary match: a proposal to increase the salary flexibility of the national resident matching program," *Journal of Economic Behavior & Organization*, vol. 66, no. 2, pp. 149–160, 2008.

[134] M. Malawski, A. Wieczorek, and H. Sosnowska, "Competition and cooperation. game theory in economy and social science," *PWN, Warszawa*, 1997.

[135] K. Basu, "The traveler's dilemma: Paradoxes of rationality in game theory," *The American Economic Review*, vol. 84, no. 2, pp. 391–395, 1994.

[136] J. G. March, *Primer on decision making: How decisions happen.* Simon and Schuster, 1994.

[137] C. F. Camerer, *Behavioral game theory: Experiments in strategic interaction.* Princeton university press, 2011.

[138] A. M. Colman, "Cooperation, psychological game theory, and limitations of rationality in social interaction," *Behavioral and brain sciences*, vol. 26, no. 2, pp. 139–153, 2003.

[139] D. M. Kreps, *Game theory and economic modelling.* Oxford University Press, 1990.

[140] J. M. Wooldridge, *Introductory econometrics: A modern approach.* Cengage learning, 2015.

[141] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman, *The elements of statistical learning: data mining, inference, and prediction*, vol. 2. Springer, 2009.

[142] M. A. Poole and P. N. O'Farrell, "The assumptions of the linear regression model," *Transactions of the Institute of British Geographers*, pp. 145–158, 1971.

[143] J. O. Rawlings, S. G. Pantula, and D. A. Dickey, *Applied regression analysis: a research tool.* Springer, 1998.

[144] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley interdisciplinary reviews: computational statistics*, vol. 2, no. 4, pp. 433–459, 2010.

[145] H. Abdi, "Partial least square regression (pls regression)," *Encyclopedia for research methods for the social sciences*, vol. 6, no. 4, pp. 792–795, 2003.

[146] S. Letzgus, P. Wagner, J. Lederer, W. Samek, K.-R. Müller, and G. Montavon, "Toward explainable ai for regression models," *arXiv preprint arXiv:2112.11407*, 2021.

[147] E. Y. Boateng and D. A. Abaye, "A review of the logistic regression model with emphasis on medical research," *Journal of data analysis and information processing*, vol. 7, no. 4, pp. 190–207, 2019.

[148] J. Tucker and D. J. Tucker, "Neural networks versus logistic regression in financial modelling: A methodological comparison," in *in Proceedings of the 1996 World First Online Workshop on Soft Computing (WSC1*, Citeseer, 1996.

[149] L. M. Healy, "Logistic regression: An overview," *Eastern Michighan College of Technology*, 2006.

[150] D. McFadden, “Regression-based specification tests for the multinomial logit model,” *Journal of econometrics*, vol. 34, no. 1-2, pp. 63–82, 1987.

[151] P. Allison, “What’s the best r-squared for logistic regression,” *Statistical Horizons*, vol. 13, 2013.

[152] N. Bussmann, P. Giudici, D. Marinelli, and J. Papenbrock, “Explainable ai in fintech risk management,” *Frontiers in Artificial Intelligence*, vol. 3, p. 26, 2020.

[153] M. E. Shipe, S. A. Deppen, F. Farjah, and E. L. Grogan, “Developing prediction models for clinical use using logistic regression: an overview,” *Journal of thoracic disease*, vol. 11, no. Suppl 4, p. S574, 2019.

[154] P. Ranganathan, C. Pramesh, and R. Aggarwal, “Common pitfalls in statistical analysis: logistic regression,” *Perspectives in clinical research*, vol. 8, no. 3, p. 148, 2017.

[155] S. C. Bankes, “Agent-based modeling: A revolution?,” *Proceedings of the National Academy of Sciences*, vol. 99, no. suppl_3, pp. 7199–7200, 2002.

[156] J. H. Miller and S. E. Page, *Complex adaptive systems: an introduction to computational models of social life: an introduction to computational models of social life*. Princeton university press, 2009.

[157] D. Richards, “Nonlinear dynamics in games: convergence and stability in international environmental agreements,” *Political Complexity: Nonlinear Models of Politics*, pp. 173–206, 2000.

[158] K. Kollman, J. H. Miller, and S. E. Page, “Adaptive parties in spatial elections,” *American Political Science Review*, vol. 86, no. 4, pp. 929–937, 1992.

[159] J. Bednar, A. Bramson, A. Jones-Rooy, and S. Page, “Emergent cultural signatures and persistent diversity: A model of conformity and consistency,” *Rationality and Society*, vol. 22, no. 4, pp. 407–444, 2010.

[160] T. C. Schelling, *Micromotives and macrobehavior*. New York: Norton, 1978.

[161] R. Axelrod, *The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration: Agent-Based Models of Competition and Collaboration*. Princeton university press, 1997.

[162] M. Laver and E. Sergenti, *Party competition: An agent-based model*, vol. 18. Princeton University Press, 2011.

[163] S. De Marchi and S. E. Page, "Agent-based models," *Annual Review of political science*, vol. 17, pp. 1–20, 2014.

[164] M. Gell-Mann, *The Quark and the Jaguar: Adventures in the Simple and the Complex*. Macmillan, 1995.

[165] M. Mitchell, *Complexity: A guided tour*. Oxford university press, 2009.

[166] A. Ghavidel and P. Pazos-Lago, "Using supervised feature selection methods to improve the predictive performance of clinical outcomes in intensive care units," in *IISE Annual Conference and Expo*, IISE, 2023.

[167] A. Ghavidel, P. Pazos, R. D. A. Suarez, and A. Atashi, "Predicting the need for cardiovascular surgery: A comparative study of machine learning models," *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 6, no. 2, pp. 92–106, 2024.

[168] R. L. Axtell and J. D. Farmer, "Agent-based modeling in economics and finance: Past, present, and future," *Journal of Economic Literature*, 2022.

[169] A. J. McLane, C. Semeniuk, G. J. McDermid, and D. J. Marceau, "The role of agent-based models in wildlife ecology and management," *Ecological modelling*, vol. 222, no. 8, pp. 1544–1556, 2011.

[170] F. Bianchi and F. Squazzoni, "Agent-based models in sociology," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 7, no. 4, pp. 284–306, 2015.

[171] E. Hunter, B. Mac Namee, and J. D. Kelleher, “A taxonomy for agent-based models in human infectious disease epidemiology,” *Journal of Artificial Societies and Social Simulation*, vol. 20, no. 3, 2017.

[172] G. Grigoryan, S. Etemadidavan, and A. J. Collins, “Computerized agents versus human agents in finding core coalition in glove games,” *Simulation*, vol. 98, no. 9, pp. 807–821, 2022.

[173] V. Grimm and S. F. Railsback, *Individual-Based Modeling and Ecology*. Princeton University Press, 2005.

[174] F. Brauer, C. Castillo-Chavez, and C. Castillo-Chavez, *Mathematical Models in Population Biology and Epidemiology*, vol. 2. Springer, 2012.

[175] U. Wilensky and W. Rand, *An Introduction to Agent-Based Modeling: Modeling Natural, Social, and Engineered Complex Systems with NetLogo*. MIT Press, 2015.

[176] B. Blasius, L. Rudolf, G. Weithoff, U. Gaedke, and G. F. Fussmann, “Long-term cyclic persistence in an experimental predator–prey system,” *Nature*, vol. 577, no. 7789, pp. 226–230, 2020.

[177] G. Grigoryan and A. J. Collins, “Feature importance for uncertainty quantification in agent-based modeling,” in *2023 Winter Simulation Conference (WSC)*, pp. 233–242, IEEE, 2023.

[178] F. E. Ritter, M. J. Schoelles, K. S. Quigley, and L. C. Klein, “Determining the number of simulation runs: Treating simulations as theories by not sampling their behavior,” *Human-in-the-Loop Simulations: Methods and Practice*, pp. 97–116, 2011.

[179] P. Windrum, G. Fagiolo, and A. Moneta, “Empirical validation of agent-based models: Alternatives and prospects,” *Journal of Artificial Societies and Social Simulation*, vol. 10, no. 2, p. 8, 2007.

[180] A. Ghavidel and P. Pazos, “Machine learning (ml) techniques to predict breast cancer in imbalanced datasets: a systematic review,” *Journal of Cancer Survivorship*, pp. 1–25, 2023.

[181] F. Cima, P. Pazos, and A. M. Canto, “Internal drivers of competitiveness for micro and small software development companies in south-eastern mexico,” *International Journal of Technological Learning, Innovation and Development*, vol. 10, no. 2, pp. 176–194, 2018.

[182] O. Asiyanbola, J. Mohanty, I. Khantouti, A. Akinwale, K. Ahenkora-Doudu, R. Imam, and R. Toukebri, “Analytical outlook of the commercial space industry for the last frontier: An entrepreneurial potential evaluation of the african space sector.,” in *Proceedings of the International Astronautical Congress, IAC*, pp. IAC–19_E6_3_2_x51395, 2019.

[183] D. Wischert, P. Baranwal, S. Bonnart, M. C. Álvarez, R. Colpari, M. Daryabari, S. Desai, S. Dhoju, G. Fajardo, B. Faldu, *et al.*, “Conceptual design of a mars constellation for global communication services using small satellites,” in *Proceedings of the International Astronautical Congress, IAC*, vol. 2020, International Astronautical Federation, IAF, 2020.

[184] S. Choi, S. Mousavi, P. Si, H. G. Yhdego, F. Khadem, and F. Afghah, “Ecgbert: Understanding hidden language of ecgs with self-supervised representation learning,” *arXiv preprint arXiv:2306.06340*, 2023.

[185] H. Yhdego, M. Audette, and C. Paolini, “Fall detection using self-supervised pre-training model,” in *2022 Annual Modeling and Simulation Conference (ANNSIM)*, pp. 361–371, IEEE, 2022.

[186] R. G. Sargent, “Validation and verification of simulation models,” in *Proceedings of the 2004 Winter Simulation Conference* (S. Jain, R. Creasey, J. Himmelspach, K. White, and M. Fu, eds.), vol. 1, (Piscataway, New Jersey), pp. 183–198, Institute of Electrical and Electronics Engineers, Inc., 2004.

[187] R. Ghanem, D. Higdon, and H. Owhadi, *Handbook of Uncertainty Quantification*, vol. 6. Springer, 2017.

[188] E. Begoli, T. Bhattacharya, and D. Kusnezov, "The need for uncertainty quantification in machine-assisted medical decision making," *Nature Machine Intelligence*, vol. 1, no. 1, pp. 20–23, 2019.

[189] A. M. Ali, M. E. Shafiee, and E. Z. Berglund, "Agent-based modeling to simulate the dynamics of urban water supply: Climate, population growth, and water shortages," *Sustainable Cities and Society*, vol. 28, pp. 420–434, 2017.

[190] F. Javadnejad, M. R. Sharifi, M. H. Basiri, and B. Ostadi, "Optimization model for maintenance planning of loading equipment in open pit mines," *European Journal of Engineering and Technology Research*, vol. 7, no. 5, pp. 94–101, 2022.

[191] E. Bruch and J. Atwell, "Agent-based models in empirical social research," *Sociological Methods & Research*, vol. 44, no. 2, pp. 186–221, 2015.

[192] S. Marino, I. B. Hogue, C. J. Ray, and D. E. Kirschner, "A methodology for performing global uncertainty and sensitivity analysis in systems biology," *Journal of Theoretical Biology*, vol. 254, no. 1, pp. 178–196, 2008.

[193] E. Štrumbelj and I. Kononenko, "Explaining prediction models and individual predictions with feature contributions," *Knowledge and information systems*, vol. 41, no. 3, pp. 647–665, 2014.

[194] J. Faraway, "Linear models with r. crc press," *Boca Raton, Florida*, 2014.

[195] B. Becker and R. Kohavi, "Adult." UCI Machine Learning Repository, 1996. DOI: https://doi.org/10.24432/C5XW20.

[196] H. Kaneko, "Cross-validated permutation feature importance considering correlation between features," *Analytical Science Advances*, vol. 3, no. 9-10, pp. 278–287, 2022.

[197] S. Guiaşu, "Weighted entropy," *Reports on Mathematical Physics*, vol. 2, no. 3, pp. 165–179, 1971.

[198] P.-H. Tsui, "Ultrasound detection of scatterer concentration by weighted entropy," *Entropy*, vol. 17, no. 10, pp. 6598–6616, 2015.

[199] S. Guiasu, "Grouping data by using the weighted entropy," *Journal of Statistical Planning and Inference*, vol. 15, pp. 63–69, 1986.

[200] X. Li, X. Jia, T. Shen, M. Wang, G. Yang, H. Wang, Q. Sun, M. Wan, and S. Zhang, "Ultrasound entropy imaging for detection and monitoring of thermal lesion during microwave ablation of liver," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 8, pp. 4056–4066, 2022.

[201] W. Qu, J. Li, W. Song, X. Li, Y. Zhao, H. Dong, Y. Wang, Q. Zhao, and Y. Qi, "Entropy-weight-method-based integrated models for short-term intersection traffic flow prediction," *Entropy*, vol. 24, no. 7, p. 849, 2022.

[202] A. Zien, N. Krämer, S. Sonnenburg, and G. Rätsch, "The feature importance ranking measure," in *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2009, Bled, Slovenia, September 7-11, 2009, Proceedings, Part II 20*, pp. 694–709, Springer, 2009.

[203] M. Wojtas and K. Chen, "Feature importance ranking for deep learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 5105–5114, 2020.

[204] A. Amalanathan and S. M. Anouncia, "A review on user influence ranking factors in social networks," *International Journal of Web Based Communities*, vol. 12, no. 1, pp. 74–83, 2016.

[205] R. Ruiz, J. S. Aguilar-Ruiz, J. C. Riquelme, and N. Díaz-Díaz, "Analysis of feature rankings for classification," in *Advances in Intelligent Data Analysis VI: 6th International Symposium on Intelligent Data Analysis, IDA 2005, Madrid, Spain, September 8-10, 2005. Proceedings 6*, pp. 362–372, Springer, 2005.

[206] C. Rudin, “Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead,” *Nature machine intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

[207] J. Tang, S. Alelyani, and H. Liu, “Feature selection for classification: A review,” *Data classification: Algorithms and applications*, p. 37, 2014.

[208] K. Liu, Y. Fu, L. Wu, X. Li, C. Aggarwal, and H. Xiong, “Automated feature selection: A reinforcement learning perspective,” *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[209] C. H. Shelden, “Prevention, the only cure for head injuries resulting from automobile accidents,” *Journal of the American Medical Association*, vol. 159, no. 10, pp. 981–986, 1955.

[210] J. M. Wooldridge, M. Wadud, and J. Lye, *Introductory econometrics: Asia pacific edition with online study tools 12 months.* Cengage AU, 2016.

[211] A. Datta, S. Sen, and Y. Zick, “Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems,” in *2016 IEEE symposium on security and privacy (SP)*, pp. 598–617, IEEE, 2016.

[212] D. Castelvecchi, “Can we open the black box of ai?,” *Nature News*, vol. 538, no. 7623, p. 20, 2016.

[213] G. Grigoryan and A. J. Collins, “Is explainability always necessary? discussion on explainable ai.” `https://digitalcommons.odu.edu/cgi/viewcontent.cgiarticle=103&context=msvcapstone`, April 2022.

[214] C.-P. Tsai, C.-K. Yeh, and P. Ravikumar, “Faith-shap: The faithful shapley interaction index,” *Journal of Machine Learning Research*, vol. 24, no. 94, pp. 1–42, 2023.

[215] A. Kalousis, J. Prados, and M. Hilario, “Stability of feature selection algorithms,” in *Fifth IEEE International Conference on Data Mining (ICDM’05)*, pp. 8–pp, IEEE, 2005.

[216] S. Nogueira, K. Sechidis, and G. Brown, "On the stability of feature selection algorithms.," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 6345–6398, 2017.

[217] U. M. Khaire and R. Dhanalakshmi, "Stability of feature selection algorithm: A review," *Journal of King Saud University-Computer and Information Sciences*, 2019.

[218] W. W. B. Goh and L. Wong, "Evaluating feature-selection stability in next-generation proteomics," *Journal of bioinformatics and computational biology*, vol. 14, no. 05, p. 1650029, 2016.

[219] K. Bhattarai, "Empty core in a coalition: Why no constitution in nepal," *Indian Journal of Economics and Business*, vol. 10, no. 1, pp. 119–126, 2011.

[220] T. Solymosi and T. E. Raghavan, "An algorithm for finding the nucleolus of assignment games," *International Journal of Game Theory*, vol. 23, pp. 119–143, 1994.

[221] M. Benedek, J. Fliege, and T.-D. Nguyen, "Finding and verifying the nucleolus of cooperative games," *Mathematical Programming*, vol. 190, no. 1-2, pp. 135–170, 2021.

[222] W. Ogryczak, "On the lexicographic minimax approach to location problems," *European Journal of Operational Research*, vol. 100, no. 3, pp. 566–585, 1997.

[223] A. Cotter, M. Gupta, and H. Narasimhan, "On making stochastic classifiers deterministic," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[224] E. Diana, W. Gill, M. Kearns, K. Kenthapadi, and A. Roth, "Minimax group fairness: Algorithms and experiments," in *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 66–76, 2021.

[225] J. Abernethy, P. Awasthi, M. Kleindessner, J. Morgenstern, C. Russell, and J. Zhang, "Active sampling for min-max fairness," *arXiv preprint arXiv:2006.06879*, 2020.

[226] A. Karczmarz, A. Mukherjee, P. Sankowski, and P. Wygocki, "Improved feature importance computations for tree models: Shapley vs. banzhaf," *arXiv preprint arXiv:2108.04126*, 2021.

[227] J. Sun, G. Zhong, K. Huang, and J. Dong, "Banzhaf random forests: Cooperative game theory based random forests with consistency," *Neural Networks*, vol. 106, pp. 20–29, 2018.

[228] C. Mio *et al.*, "A power index based frameworkfor feature selection problems," 2020.

[229] N. Patel, M. Strobel, and Y. Zick, "High dimensional model explanations: An axiomatic approach," in *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pp. 401–411, 2021.

[230] B. Kulynych and C. Troncoso, "Feature importance scores and lossless feature pruning using banzhaf power indices," *arXiv preprint arXiv:1711.04992*, 2017.

[231] K. Fotion, "Feature power: a new variable importance measure for random forests," 2018.

[232] D. Vernon-Bido, G. Grigoryan, H. Kavak, and J. Padilla, "Assessing the impact of cyberloafing on cyber risk," in *Proceedings of the Annual Simulation Symposium*, pp. 1–9, 2018.

[233] A. Ghavidel, R. Ghousi, and A. Atashi, "An ensemble data mining approach to discover medical patterns and provide a system to predict the mortality in the icu of cardiac surgery based on stacking machine learning method," *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, pp. 1–11, 2022.

[234] A. J. Collins and G. Grigoryan, "Abmscore: a heuristic algorithm for forming strategic coalitions in agent-based simulation," *Journal of Simulation*, pp. 1–25, 2024.

[235] D. P. Schultz, "The human subject in psychological research.," *Psychological bulletin*, vol. 72, no. 3, p. 214, 1969.

[236] M. Chromik and M. Schuessler, "A taxonomy for human subject evaluation of black-box explanations in xai.," *Exss-atec@ iui*, vol. 1, 2020.

[237] J. Heo, S. Joo, and T. Moon, "Fooling neural network interpretations via adversarial model manipulation," *Advances in neural information processing systems*, vol. 32, 2019.

[238] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim, "Sanity checks for saliency maps," *Advances in neural information processing systems*, vol. 31, 2018.

[239] S. Hooker, D. Erhan, P.-J. Kindermans, and B. Kim, "Evaluating feature importance estimates," *arXiv preprint arXiv:1806.10758*, vol. 2, 2018.

[240] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, *et al.*, "Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav)," in *International conference on machine learning*, pp. 2668–2677, PMLR, 2018.

[241] H. Lakkaraju, S. H. Bach, and J. Leskovec, "Interpretable decision sets: A joint framework for description and prediction," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1675–1684, 2016.

[242] P. Sprent and N. C. Smeeton, *Applied nonparametric statistical methods.* CRC press, 2016.

[243] C. Molnar, T. Freiesleben, G. König, J. Herbinger, T. Reisinger, G. Casalicchio, M. N. Wright, and B. Bischl, "Relating the partial dependence plot and permutation feature importance to the data generating process," in *World Conference on Explainable Artificial Intelligence*, pp. 456–479, Springer, 2023.

[244] A. Fisher, C. Rudin, and F. Dominici, "All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously.," *J. Mach. Learn. Res.*, vol. 20, no. 177, pp. 1–81, 2019.

[245] H. Ishwaran and M. Lu, "Standard errors and confidence intervals for variable importance in random forest regression, classification, and survival," *Statistics in medicine*, vol. 38, no. 4, pp. 558–582, 2019.

[246] S. Janitza, E. Celik, and A.-L. Boulesteix, "A computationally fast variable importance test for random forests for high-dimensional data," *Advances in Data Analysis and Classification*, vol. 12, pp. 885–915, 2018.

[247] J. Lin, “Divergence measures based on the shannon entropy,” *IEEE Transactions on Information theory*, vol. 37, no. 1, pp. 145–151, 1991.

[248] D. Ellerman and D. Ellerman, “The relationship between logical entropy and shannon entropy,” *New Foundations for Information Theory: Logical Entropy and Shannon Entropy*, pp. 15–22, 2021.

[249] B. D. Sharma, J. Mitter, and M. Mohan, “On measures of “useful” information,” *Information and Control*, vol. 39, no. 3, pp. 323–336, 1978.

[250] P. K. Newton and S. A. DeSalvo, “The shannon entropy of sudoku matrices,” *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 466, no. 2119, pp. 1957–1975, 2010.

[251] B. Fadlallah, B. Chen, A. Keil, and J. Príncipe, “Weighted-permutation entropy: A complexity measure for time series incorporating amplitude information,” *Physical Review E*, vol. 87, no. 2, p. 022911, 2013.

[252] P. Sedgwick, “Spearman’s rank correlation coefficient,” *Bmj*, vol. 349, 2014.

[253] A. L. Edwards, “An introduction to linear regression and correlation,” *The Correlation Coefficient*, pp. 33–46, 1976.

[254] D. Mohamad, B. M. Deros, D. A. Wahab, D. D. Daruis, and A. R. Ismail, “Integration of comfort into a driver’s car seat design using image analysis,” *American Journal of Applied Sciences*, vol. 7, no. 7, p. 937, 2010.

[255] F. Hsieh, P. W. Lavori, H. J. Cohen, and J. R. Feussner, “An overview of variance inflation factors for sample-size calculation,” *Evaluation & the Health Professions*, vol. 26, no. 3, pp. 239–257, 2003.

[256] F. Song, Z. Guo, and D. Mei, “Feature selection using principal component analysis,” in

2010 international conference on system science, engineering design and manufacturing informatization, vol. 1, pp. 27–30, IEEE, 2010.

[257] Q. Guo, W. Wu, D. Massart, C. Boucon, and S. de Jong, "Feature selection in principal component analysis of analytical data," *Chemometrics and Intelligent Laboratory Systems*, vol. 61, no. 1-2, pp. 123–132, 2002.

[258] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers & Electrical Engineering*, vol. 40, no. 1, pp. 16–28, 2014.

[259] J. Sanchez-Soriano, "An overview on game theory applications to engineering," *International Game Theory Review*, vol. 15, no. 03, p. 1340019, 2013.

[260] S. Das, "Filters, wrappers and a boosting-based hybrid for feature selection," in *Icml*, vol. 1, pp. 74–81, Citeseer, 2001.

[261] R. B. Bendel and A. A. Afifi, "Comparison of stopping rules in forward "stepwise" regression," *Journal of the American Statistical association*, vol. 72, no. 357, pp. 46–53, 1977.

[262] T. N. Lal, O. Chapelle, J. Weston, and A. Elisseeff, "Embedded methods," in *Feature extraction: Foundations and applications*, pp. 137–165, Springer, 2006.

[263] M. Balinski and R. Laraki, *Majority judgment: measuring, ranking, and electing.* MIT press, 2011.

[264] M. Grabisch, "The core of games on ordered structures and graphs," *Annals of Operations Research*, vol. 204, no. 1, pp. 33–64, 2013.

[265] B. Hu, J. Shao, and M. Palta, "Pseudo-r 2 in logistic regression model," *Statistica Sinica*, pp. 847–860, 2006.

VITA

Gayane Grigoryan
Department
Old Dominion University
Norfolk, VA 23529

Gayane Grigoryan earned her Master's degree in Economics from Old Dominion University in 2017 and her Bachelor's degree in Business Administration and Entrepreneurship from Armenian State University of Economics in 2012. Gayane's research interests include explainable artificial intelligence, machine learning, human-centered analysis, large-scale language models, and time series data analysis applied to healthcare and cybersecurity systems. She published journal and conference papers in Winter Simulation Conference, International Journal of Systems of Systems Engineering, Journal of Simulation, Spring Simulation Conference, and ACM SIGSIM International Conference on Principles of Advanced Discrete Simulation. Her email address is grigory.gayaneh@gmail.com. For details about her work, please check the website `https://grigoryangayane.com/`.

Typeset using LaTeX.