UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

DANTE DE ARAÚJO COSTA

Uma Abordagem de Aprendizado de Máquina comparando o Desempenho Preditivo e a Interpretabilidade de Modelos para Prever o Sucesso de Jogadores de Basquete da NCAA em Alcançar a NBA

CAMPINA GRANDE - PB

2024

DANTE DE ARAÚJO COSTA

UMA ABORDAGEM DE APRENDIZADO DE MÁQUINA COMPARANDO O DESEMPENHO PREDITIVO E A INTERPRETABILIDADE DE MODELOS PARA PREVER O SUCESSO DE JOGADORES DE BASQUETE DA NCAA EM ALCANÇAR A NBA

Trabalho de Conclusão Curso apresentado ao Curso Bacharelado em Ciência da Computação do Centro de Engenharia Elétrica e Informática da Universidade Federal de Campina Grande, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

Orientadora : Joseana Macêdo Fechine

CAMPINA GRANDE - PB

2024

DANTE DE ARAÚJO COSTA

UMA ABORDAGEM DE APRENDIZADO DE MÁQUINA COMPARANDO O DESEMPENHO PREDITIVO E A INTERPRETABILIDADE DE MODELOS PARA PREVER O SUCESSO DE JOGADORES DE BASQUETE DA NCAA EM ALCANÇAR A NBA

Trabalho de Conclusão Curso apresentado ao Curso Bacharelado em Ciência da Computação do Centro de Engenharia Elétrica e Informática da Universidade Federal de Campina Grande, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

BANCA EXAMINADORA:

Joseana Macêdo Fechine – UASC/CEEI/UFCG

Patrícia Duarte de Lima Machado
Examinador – UASC/CEEI/UFCG

Francisco Vilar Brasileiro
Professor da Disciplina TCC – UASC/CEEI/UFCG

Trabalho aprovado em: 17 de Maio de 2024.

CAMPINA GRANDE - PB

RESUMO

Modelos preditivos em aprendizado de máquina e processos de descoberta de conhecimento em bases de dados, particularmente em domínios como o basquete, são inestimáveis para obter insights sobre o desempenho dos jogadores. Este estudo compara abordagens de aprendizado de máquina supervisionado (modelos de caixa preta e caixa branca, incluindo métodos de conjunto) para analisar dados estatísticos de jogadores de basquete universitário (NCAA). Nosso objetivo é identificar jogadores da NCAA com alto potencial para sucesso na NBA, determinar quais características dos jogadores mais influenciam as decisões de seleção e como esses modelos chegam a tais conclusões para comparar seus desempenhos e a explicabilidade associada. Esta tarefa é desafiadora devido a fatores além das estatísticas, como o contexto do jogador e as considerações do elenco da equipe durante a seleção. O objetivo principal é fornecer aos tomadores de decisão insights cruciais para a seleção de jogadores, ajudar na melhor avaliação de jogadores e desenvolver jovens talentos enfatizando aspectos-chave do jogo. Comparamos os resultados de modelos de predição interpretáveis com níveis satisfatórios de precisão. Equilibrando interpretabilidade e precisão preditiva, empregamos métodos de classificação de caixa branca, caixa preta e de conjunto, como Árvores de Decisão, Regressão Logística, Máquina de Vetores de Suporte, Perceptron Multicamadas, Floresta Aleatória e XGBoost. Além disso, algoritmos genéticos foram usados para reduzir o conjunto de características de cada modelo, retendo apenas as características mais impactantes. Comparado aos procedimentos padrão sem seleção de características, todos os modelos mostraram desempenho melhorado. Encontramos diferenças mínimas na precisão preditiva entre os melhores modelos de caixa branca e caixa preta. A combinação de algoritmos genéticos e regressão logística superou a precisão preditiva de outros modelos, reduzindo significativamente as características e melhorando a interpretabilidade dos resultados. A análise também destaca as características mais influentes no modelo e como os modelos chegaram a tais conclusões.

A MACHINE LEARNING APPROACH COMPARING PREDICTIVE PERFORMANCE AND INTERPRETABILITY OF MODELS FOR PREDICTING SUCCESS OF NCAA BASKETBALL PLAYERS TO REACH NBA

ABSTRACT

Predictive models in machine learning and knowledge discovery in database processes, particularly in domains like basketball, are invaluable for gaining insights into player performance. This study compares supervised machine learning approaches (black-box and white-box models, including ensemble methods) to analyze statistical data from college basketball players (NCAA). We aim to identify NCAA players with high potential for NBA success, determine which player characteristics most influence selection decisions, and how these models have such conclusions to compare their performances and the associated explainability. This task is challenging due to factors beyond statistics, such as player context and team roster considerations during selection. The main objective is to provide decision-makers with crucial insights for player selection, aid in better player assessment, and develop young talents by emphasizing key game aspects. We compare interpretable prediction model results with satisfactory accuracy levels. Balancing interpretability and predictive accuracy, we employ white-box, black-box, and ensemble classification methods like Decision Trees, Logistic Regression, Support Vector Machine, Multi-Layer Perceptron, Random Forest, and XGBoost. Additionally, genetic algorithms were used to reduce each model's feature set, retaining only the most impactful features. Compared to standard procedures without feature selection, all models showed improved performance. We found minimal differences in predictive accuracy between the best white-box and black-box models. Genetic algorithms and logistic regression combination outperformed other models' predictive accuracy while significantly reducing features and enhancing result interpretability. The analysis also highlights the most influential features in the model and how models came to such conclusions.

A Machine Learning Approach comparing predictive performance and Interpretability of models for Predicting Success of NCAA Basketball Players to Reach NBA

Dante de Araujo Costa
Department of Systems and Computing
Federal University of Campina Grande
Campina Grande, Brazil
dante.costa@ccc.ufcg.edu.br

Joseana Macêdo Fechine
Department of Systems and Computing
Federal University of Campina Grande
Campina Grande, Brazil
joseana@dsc.ufcg.edu.br

ABSTRACT

Predictive models in machine learning and knowledge discovery in database processes, particularly in domains like basketball, are invaluable for gaining insights into player performance. This study compares supervised machine learning approaches (black-box and white-box models, including ensemble methods) to analyze statistical data from college basketball players (NCAA). We aim to identify NCAA players with high potential for NBA success, determine which player characteristics most influence selection decisions, and how these models have such conclusions to compare their performances and the associated explainability. This task is challenging due to factors beyond statistics, such as player context and team roster considerations during selection. The main objective is to provide decision-makers with crucial insights for player selection, aid in better player assessment, and develop young talents by emphasizing key game aspects. We compare interpretable prediction model results with satisfactory accuracy levels. Balancing interpretability and predictive accuracy, we employ white-box, black-box, and ensemble classification methods like Decision Trees, Logistic Regression, Support Vector Machine, Multi-Layer Perceptron, Random Forest, and XGBoost. Additionally, genetic algorithms were used to reduce each model's feature set, retaining only the most impactful features. Compared to standard procedures without feature selection, all models showed improved performance. We found minimal differences in predictive accuracy between the best white-box and black-box models. Genetic algorithms and logistic regression combination outperformed other models' predictive accuracy while significantly reducing features and enhancing result interpretability. The analysis also highlights the most influential features in the model and how models came to such conclusions.

Keywords

Predictive models, Machine Learning, Feature Selection, Genetic Algorithms, Interpretability.

1. INTRODUCTION

Decades of data collection exist for American basketball leagues, but the widespread adoption of Artificial Intelligence (AI) applied to performance improvement in this domain has occurred only in recent years. Researchers now use data mining and machine learning techniques to uncover factors that scouts and other sports professionals may not immediately notice but can lead to success with appropriate training.

This study applies the Knowledge Discovery in Databases (KDD) process, mainly on machine learning algorithms, to data and features from the National Collegiate Athletic Association (NCAA) men's basketball datasets. The goal is to identify players with the best odds of succeeding professionally and understand why a machine learning algorithm would recommend a particular athlete.

The NBA Draft is an annual event where NBA teams select eligible players to join their rosters. It consists of two rounds, with 60 players chosen in total. Each team has one pick per round, with the draft order determined by a lottery system based on teams' records from the previous season. Players typically declare their intention to enter the draft after completing their college eligibility.

Utilizing statistical data from NCAA matches is cost-effective compared to other techniques like computer vision or scouting. This method allows for analyzing every team and athlete in the league since the data is collected after each match. The study aims to automatically identify NCAA basketball players with a good chance of reaching the NBA by applying supervised machine learning techniques. Various classification methods, including induction of Decision Trees (C4.5, C5.0, and CART algorithms), Logistic Regression, Support Vector Machine, and Multi-layer Perceptron (MLP), are used to compare predictive accuracy and comprehensibility between black-box and white-box models, as well as ensemble models, like Random Forest and XGBoost.

The study's dataset contains redundant and irrelevant features that could negatively impact decision-making. To address this, genetic algorithms are used in the feature selection process to filter player attributes that contribute the most to being chosen by an NBA team. This approach aims to improve the predictive

accuracy of the models while reducing the number of features needed for the models to explain the decision-making process.

The purpose of this study is to do an investigation process considering the following research questions:

• **RQ1** - Which classification techniques, among those with reasonable accuracy, provide better explanations for decisions?

• **RQ2** - Which classification techniques, among those studied, exhibit the best predictive performance?

• **RQ3** - Is there any model that provides a solution in the domain that offers high quality in both accuracy and explainability?

2. BACKGROUND

Most existing work on using machine learning models to predict basketball performance has been conducted in a statistical context. These efforts primarily focus on predicting player performance and match outcomes in leagues such as the NCAA and NBA. The work discussed here focuses explicitly on predicting the success of NCAA basketball players in reaching the NBA, using statistical data from their college careers. More specifically, our approach emphasizes achieving high prediction accuracy while also considering the interpretability of the models. Therefore, this section provides some background knowledge to help readers understand the key concepts related to supervised machine learning techniques.

2.1 Supervised Machine Learning

In this study, we invested in a sample of techniques that includes induction of Decision Trees, Logistic Regression as single and white-box models, MLP and SVM as single and black-box models, and Random Forest and XGBoost as ensemble methods.

2.1.1 Decision Trees

Induction of decision trees is a supervised learning method used for classification and regression tasks. They partition the feature space into regions and make predictions based on the majority class (for classification) or the average value (for regression) of the training examples within each region. More specifically, decision trees recursively partition the feature space into regions based on feature values, each representing a decision node. The tree selects the feature and split point at each decision node that best separates the data into different classes or values, typically using metrics like Gini impurity, entropy, and information gain. Decision trees are easy to interpret and visualize, making them popular for understanding and explaining data. Among the main algorithms for building decision trees are Classification And Regression Trees (CART) [1] and C4.5 and C5.0, developed by Ross Quinlan [2].

2.1.2 Logistic Regression

It is a supervised learning algorithm used for binary classification tasks. It utilizes the logistic (or sigmoid) function to transform a linear combination of input features into a probability value between 0 and 1 [3]. This probability indicates the likelihood that a given input corresponds to one of two predefined categories. The essential mechanism of Logistic Regression is grounded in the logistic function's ability to accurately model the probability of binary outcomes.

2.1.3 Support Vector Machines

SVM is a supervised learning algorithm used for classification and regression tasks. SVM works by finding the hyperplane that best separates the data points of different classes [4]. The hyperplane is chosen to maximize the margin, i.e., the distance between the hyperplane and the nearest data points from each class, also known as support vectors. SVM can handle linear and non-linearly separable data using kernel functions such as linear, polynomial, radial basis function (RBF), etc.

2.1.4 Multi-Layer Perceptron

MLP is an artificial neural network consisting of at least three layers of nodes: an input layer, one or more hidden layers, and an output layer [5]. Each node, or neuron, in one layer, is connected to every node in the subsequent layer, and each connection has an associated weight. Each node uses a nonlinear activation function, with sigmoid and ReLU functions commonly used. MLPs can learn nonlinear models and are trained using the backpropagation method. The MLP was a crucial development in the history of neural networks.

2.1.5 Random Forest

It is an ensemble learning method that combines multiple decision trees to improve predictive performance [6]. This combination occurs during training and outputs the mode of the classes (for classification) or the average prediction (for regression) of the individual trees. Random Forest builds a forest of decision trees by repeatedly selecting random subsets of the training data and features. Each decision tree is trained on a bootstrap sample of the original dataset (sampling with replacement), and at each node, only a random subset of features is considered for splitting. During prediction, each tree in the forest independently makes a prediction, and the final output is determined by aggregating the predictions of all trees (e.g., majority voting for classification or averaging for regression).

The main parameters of a Random Forest include the number of trees in the forest, the node splitting criterion (such as Gini impurity or entropy), the maximum tree depth, and the minimum number of samples required to split a node. The appropriate choice of these parameters is crucial for the model's performance and generalization.

2.1.6 XGBoost (Extreme Gradient Boosting)

It is short for eXtreme Gradient Boosting, an optimized implementation of gradient-boosting algorithms designed for speed and performance [7]. Specifically, XGBoost is an ensemble learning technique that builds a series of decision trees sequentially, where each tree corrects the errors made by the previous ones. It uses gradient boosting, which minimizes a loss function by iteratively adding weak learners (decision trees) to the ensemble. XGBoost is highly customizable and allows for various hyperparameter tuning to optimize performance. It incorporates regularization techniques, such as shrinkage (learning rate) and pruning, to prevent overfitting. XGBoost supports regression and classification tasks and is known for its high accuracy and efficiency.

2.2 Genetic Algorithms for Attribute Reduction

The application of genetic algorithms for attribute reduction in datasets has proven significant in data mining. In an experiment conducted by Babatunde et al. [8], a genetic algorithm was used on a dataset with 100 attributes, successfully reducing the dimensionality to only 11 attributes. The study compared two methods, WEKA (Information Gain Ranking Filter) and WEKA (CFS Subset Evaluator), which managed to reduce the dataset to only 20 attributes. However, the genetic algorithm outperformed these methods by achieving a more substantial reduction to 11 attributes.

Furthermore, the author assessed the accuracy of several machine learning models using the features generated by the three models. The model that achieved the highest accuracy, at 94%, utilized the features generated by combining the genetic algorithm with a Multi-Layer Perceptron.

2.3 Explainability

Explainability is one of the hot topics in Artificial Intelligence, with the field named eXplainable Artificial Intelligence (XAI), including interpretable machine learning. In general terms, explainability and interpretability refer to the degree to which a human can understand and trust the decisions made by a machine-learning model. In supervised machine learning, where models learn from labeled data, it is crucial to know how and why the model arrives at its predictions. Here in the present work, we are interested in identifying which features are essential and how they contribute to the predictions, as well as in understanding the overall behavior of the ML model.

The existing literature often overlooks the importance of explainability associated with performance and selecting different types of features to measure the result, as each type of algorithm may return a different outcome. The intersection of these features will highlight the characteristics that deserve attention in the addressed problem. In what follows, we summarize this subject, presenting its relevant concepts and approaches.

Two categories of supervised machine learning techniques have been discussed: white-box and black-box models. In white boxes, it is assumed that the models are inherently interpretable [9], where the model form admits valuable explanations of its output without any post-processing. These models offer high explainability because their decision-making process is simple and directly tied to the input features. Examples of such models include decision trees and logistic regression. There is specific literature on developing and evaluating the performance of inherently interpretable models.

On the other hand, black boxes refer to models whose behavior is not directly understandable. They use algorithms where the relationship between inputs and outputs is not easily interpretable. Examples include MLP, SVM, XGBoost, and random forests.

There are approaches and tools primarily aimed at the interpretability of black-box models, which can also be used in white-box models. For example, SHAP and LIME are two well-known and widely used approaches, where SHAP comes from SHapley Additive exPlanations and LIME stands for Local Interpretable Model-agnostic Explanations [10]. They were primarily developed to explain predictions of black-box models. Thus, they offer valuable insights for explaining black-box machine learning models and help improve their interpretability by providing global and local explanations. However, in white-box models like decision trees and logistic regression that are already inherently interpretable, SHAP and LIME can complement their interpretability by providing additional insights into feature importance and local explanations for individual predictions.

3. RELATED WORK

The field of machine learning applied to sports, including basketball, has grown significantly in recent years, both in research and in its application within teams. In North American professional leagues like the NBA, extensive data collection and analysis are standard practices for understanding the probabilities of success for individual players and teams. Houde and Matthew [11] conducted a study comparing different models and metrics for predicting game outcomes based on data from previous seasons. The present work is focused on predictive modeling and performance analysis in basketball, mainly considering the interpretability aspects of the models.

Mahmood et al. [12] analyzed the potential for specific players to become up-and-coming stars in the NBA, using a concept called Co-player, which refers to teammates or opponents who played during a determined period. Co-player was found to be a significant factor in predicting rising stars, with machine learning algorithms such as Support Vector Machine (SVM), Decision Tree CART, Maximum Entropy Markov Model (MEM), Bayesian Network, and Naive Bayes. In the study, some new attributes were created based on the preexisting attributes in the databases. For example, the average Hollinger Score of a player's Co-players was computed. At the end of the study, the significance of these Co-player-related attributes in aiding the prediction of rising stars was demonstrated.

Meanwhile, Albert et al. [13] proposed a hybrid approach called ANN (Adaboost, Random Forest, and Multi-Layer Perceptron - MLP) that feeds on the same dataset. According to the author, this weighted combination of the three conventional models has not been the subject of research, making it an innovative approach to the problem of predicting stars in the NBA. This combination was obtained from the individual results of various tested machine learning models, and the mentioned three models yielded better metrics in terms of sensitivity and specificity. This ANN was constructed as a Recurrent Neural Network (RNN) hidden layer. Upon retesting the proposed model, the authors achieved a specificity of 90% and a sensitivity of 80%. While the specificity decreased slightly compared to the individual models, there was a significant increase in sensitivity.

Additionally, Hsu et al. [14] focused on predicting the top sixteen NBA teams by applying machine learning algorithms based on player characteristics such as points, blocks, offensive and defensive rebounds, and other game metrics. Models like Polynomial Regression, Random Forest Regression, and Support Vector Regression were employed to calculate players' winning contributions to their teams, using the player efficiency rating (PER) to measure player performance.

4. METHODOLOGY

This section presents the study design and the methods used to prepare and process data, following the traditional pipeline. Thus, it is divided into four subsections that represent the stages of the Knowledge Discovery in Databases (KDD) process that was addressed, namely: (i) Data Description, (ii) Pre-Processing, (iii) Used Algorithms, and (iv) Evaluation Metrics.

4.1 Data Description

The selected dataset contains registers from the period 2009 to 2021 of American university athletes who competed in the NCAA [19]

• The database consists of 65 features and 65.039 instances related to the athletes and the matches they played. Some of the attributes found are:

• Attributes: The minutes played per game (Min), position, field goals made (FGM), field goals attempted (FGA), 3-pointers made (3 PTM), 3-pointers attempted (3 PTA), free throws made (FTM), free throws attempted (FTA), offensive rebounds (OREB), defensive rebounds (DREB), rebounds in general (REB), assists (AST), steals (STL), blocks (BLK), personal fouls (PF), points (PTS), and starter status (Starter if true, reserve if false) are some of the attributes found in the database.

4.2 Pre-Processing

In our dataset analysis, we observed that it is unbalanced, with the distribution shown in Table 1. We plan to address this imbalance, using techniques such as resampling and weighted loss functions during model training.

Label	Number of samples in the dataset
0	64,337
1	320
2	282

Table 1: Amount of samples for each label

The label 0 denotes the number of participants not selected to play in the NBA. Label 1 represents individuals drafted in the first round, the top thirty players with higher priority. Label 2 comprises players selected in the last thirty, with lower priority than label 1. To balance the number of instances, we initially tested both Undersampling and Oversampling techniques, but they did not yield satisfactory prediction indexes. An alternative approach was adopted, involving the following steps:

- Separation of all players belonging to label 1;
- Separation of all players belonging to label 2;
- Random selection of 1990 instances from a total of 64,337.

As a result, a balanced dataset consisting of 2,592 instances was obtained for further tests. The number of cases of undrafted players was initially proposed to maintain the same proportion of players in each label as in the last draft, when 243 declared for it. However, only 60 were selected in one of the two rounds.

Pre-processing is a stage of KDD where several techniques are applied to the data to improve the learning rates of the models. Blum's work [15] emphasizes the importance of feature selection for machine learning models. This paper focuses on the genetic algorithm combined with the predictive models used.

We set the test size at 0.33, which means 33% of the data will be in the testing partition, and the other 67% will be used for the training partition.

4.3 Feature Selection Method

The proposed method can be visualized in Figure 1.

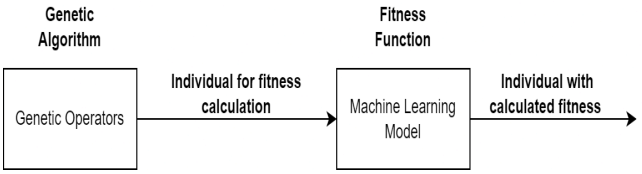

Figure 1: Method for Feature Selection

This framework utilizes a genetic algorithm, which is an algorithmic framework. An individual undergoes genetic operators in this framework, including one-point crossover, inversion mutation, and elitist selection. After these genetic operators, the chromosome is evaluated using a fitness function incorporating a machine-learning algorithm. The termination condition is then checked to determine if it has been met. If the condition is met, the output will be a subset containing the best-performing chromosomes. If not, the process is repeated.

4.4 Used Algorithms

The study[21] employed a variety of machine learning algorithms to diversify the modeling approaches. These included ensemble methods such as XGBoost and Random Forest, tree-based approaches like C4.5, C5.0, and CART, and a statistical model, Logistic Regression. Black-box algorithms such as Support Vector Machine (SVM) and Neural Networks with Multi-Layer Perceptron architecture were also utilized. These algorithms were chosen to comprehensively understand the dataset's behavior, considering performance and comprehensibility metrics in different contexts.

Configuring hyperparameters is crucial for optimizing machine learning algorithms. To identify the best hyperparameters for the dataset context, we used GridSearch. This systematic approach automates parameter-tuning by generating and evaluating various parameter combinations. The combination that best fits the dataset is the most suitable [16].

We present the best hyperparameter combinations obtained through GridSearch for each algorithm.

4.4.1 Decision Tree CART

The hyperparameters adopted for Cart algorithms are these:

• Random state: 33

• Criterion: Gini

• Max depth: None

• Max features: None

• Min samples leaf: 1

• Splitter: best

4.4.2 Decision Tree C4.5 and Decision Tree C5.0

The hyperparameters GridSearch used for the feature selection criterion were identical to those selected in the CART model. The implications of this will be shown and discussed later in the Result section.

4.4.3 Logistic Regression (LR)

In Logistic Regression, we adjusted the algorithm's hyperparameters using the following setup:

• Random state: 33

• Max iter: 500

• Solver: lbfgs

4.4.4 Support Vector Machine (SVM)

For more information on the algorithm's work, please refer to [17] (Zhou, 2021). The following hyperparameter settings were used:

• C: 0.001

• Gamma: Scale

• Kernel: Linear

4.4.5 Multi-Layer Perceptron (MLP)

The hyperparameters adopted for the MLP neural network are:

• Activation function: Rectified Linear Unit (ReLU)

• 2 Hidden layer sizes: (256 and 128 neurons, respectively)

• Learning rate init: 0.01

• Solver: Adam

• Max iter: 200

4.4.6 Random Forest

The setup adopted for the Random Forest algorithm included the following:

• Random state: 33

• Max depth: 5

• Min Samples Leaf: 2

• Max features: sqrt

• n estimators: 100

4.4.7 XGBoost

The hyperparameters adopted for the XGBoost algorithm are:

• Random state: 33

•Learning rate: 0.2

• Max depth: 5

• Objective: multi:softprob

• n estimators: 100

4.4.8 Genetic Algorithm - GA

The settings adopted were:

• individual's representation = binary

• length population = 50

• length chromosome = 13

• crossover rate = 75

• mutation rate = 30

The stopping criterion used was a counter that records the number of generations the best individual found remained unchanged. The algorithm ends if this individual is not modified for ten consecutive generations. Otherwise, the stop criterion counter will be reset, and the process will be repeated.

4.5 Test Environment

In the experiments, a Google Compute Engine instance with approximately 12 GB of RAM and 108 GB of hard drive space was used [18].

4.6 Evaluation Metrics

In this research, various metrics were employed to evaluate the models, reflecting the different nature of the models and the types of data under analysis. Different metrics were necessary to capture the nuances of each model's performance in its specific context.

4.6.1 Performance

The selected metrics for performance were Accuracy, Precision, Recall, and F1 score as comparison measures between the algorithms, aiming to evaluate the performance of each algorithm according to the input parameters of each type of feature selection.

4.6.2 Feature Relevance

For Random Forest and XGBoost models, feature importance is critical to understanding how the model makes decisions. These models assign an importance score to each feature based on how much they contribute to reducing impurity or error in the model's predictions. This score is calculated during the training process and reflects the relative importance of each feature in making accurate predictions. By analyzing the importance of features, we can identify which features have the most significant impact on the model's output and gain insights into the underlying patterns in the data.

Logistic Regression, on the other hand, calculates feature importance based on the absolute values of the coefficients assigned to each feature in the model. These coefficients represent the strength and direction of the relationship between each feature and the target variable. Logistic Regression measures each feature's overall importance in the model by taking the average of the absolute coefficients across all classes. This approach also allows us to identify the most influential features in the model and understand how they contribute to the model's predictions.

Permutation importance determines feature importance for Support Vector Machine (SVM) models. This technique evaluates the impact of each feature by randomly shuffling its values and measuring the resulting change in the model's performance. In SVMs with a linear kernel, the absolute coefficients are used directly to determine feature importance. However, for SVMs with nonlinear kernels, the average absolute coefficients across classes are considered, providing insight into the significance of each feature in the model's decision boundaries.

Permutation importance was also computed for the Multi-Layer Perceptron (MLP) model. This technique evaluates the significance of each feature by permuting its values and measuring the impact on the model's performance. Features that, when permuted, cause the most significant decrease in performance are considered more important.

By employing these techniques, we gained insights into how each model makes predictions and which features are most

influential, enhancing the overall interpretability of the machine learning models used in this study.

4.6.3 Interpretability

To evaluate the interpretability of our models, we employed the LIME (Local Interpretable Model-agnostic Explanations) technique. LIME offers detailed and accessible explanations regarding the impact of each feature on predictions, providing insights into each model's decision-making process. By operating locally and analyzing individual predictions, LIME becomes a powerful tool for understanding the rationale behind machine learning models' decisions.

While calculating feature importance provides a general overview of which features are most relevant to the model's predictions, it lacks the depth and context provided by LIME. LIME's model-agnostic nature allows it to offer local explanations for individual predictions across various model types, including Logistic Regression, SVM, and MLP models. This approach enhances our ability to understand how these models make decisions in specific instances, as it ensures a consistent procedure for every model used, contributing to the comparison of their interpretability and reliability.

Although the SHAP (Shapley Additive exPlanations) method is commonly used for models like XGBoost and Random Forest to quantify and visualize feature impacts, we opted not to use it in this study. Instead, we focused on LIME due to its ability to provide detailed and context-rich explanations for individual predictions, which is crucial for understanding the decision-making process of all models at a granular level and for facilitating comparison.

5. RESULTS

This section presents the outcomes and contributions of the Machine Learning-based Approach proposed in this study. The section is divided into three subsections to systematically describe the results of the algorithms discussed in subsection 4.4. We will begin by elaborating on the outcomes of the genetic algorithm on the database, followed by an analysis of the prediction results. Furthermore, the final subsection analyzes the best predictive models' interpretability aspects, using LIME to explore each model.

5.1 Genetic Algorithm and Feature Selection

After the feature selection process, the genetic algorithm successfully reduced the original 65 features to a subset of 28. The selected attributes are as follows: **['GP', 'Min_per', 'usg', 'eFG', 'TS_per', 'ORB_per', 'DRB_per', 'AST_per', 'FTA', 'FT_per', 'twoPM', 'twoPA', 'ftr', 'adjoe', 'pfr', 'midmade', 'midmade+midmiss', 'midmade/(midmade+midmiss)', 'dunksmade', 'drtg', 'adrtg', 'dporpag', 'obpm', 'dbpm', 'mp', 'dreb', 'stl', 'pts']**.

5.2 Algorithms and Predictive Performance

We evaluated the models using standard classification metrics, including accuracy, precision, recall, and F1-score, as shown in Table 2 and Figure 2.

As outlined in section 4.4.2, the hyperparameters for decision tree-based models were consistent across all models. The GridSearch configuration constrained the hyperparameter grid, limiting the range of combinations and potentially leading to similar outcomes for different models. This uniformity might have been different with a larger grid. However, the decision to work with fewer possibilities was justified by hardware limitations for testing in the environment described in section 4.5.

Model	Accuracy	Precision	Recall	F1 Score
Random Forest	85.98%	85.15%	85.98%	85.35%
MLP	85.75%	85.36%	85.75%	85.47%
XGBoost	83.29%	82.23%	83.29%	82.65%
SVM	82.48%	82.39%	82.48%	82.18%
LR	85.28%	84.33%	85.28%	84.76%
C4.5/C5.0 /CART	80.84%	79.89%	80.85%	80.27¨%

Table 2: Performance Metrics for the algorithms

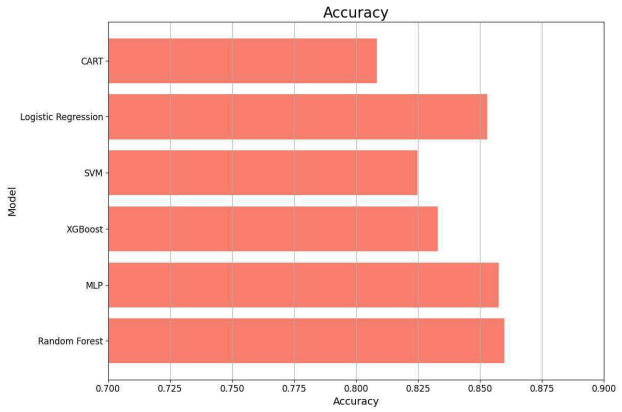

Figure 2: Accuracy comparison of models

This uniformity, feature selection process, and the inherent characteristics of these models resulted in identical predictive outcomes. Although the primary focus of the study was to evaluate the interpretability of the best-performing models (none of which were decision tree-based), we found that the format and significance were broadly similar when examining the feature importance of the trees.

Among the models tested, Random Forest and Multi-Layer Perceptron (MLP) demonstrated the highest accuracy, achieving around 0.86. This indicates their ability to correctly classify players' NBA success based on NCAA performance metrics. Random Forest also showed high precision, recall, and F1-score, indicating its effectiveness in identifying players with a high likelihood of NBA success while minimizing false positives.

Logistic Regression, SVM, and XGBoost also performed well, with approximately 0.85, 0.82, and 0.83 accuracies, respectively. These models showed competitive precision, recall, and F1-score performance, highlighting their effectiveness in predicting NBA success based on NCAA performance metrics.

The strong performance of Random Forest can be attributed to its ensemble learning nature, which combines multiple decision trees to improve prediction accuracy. MLP, on the other hand, is

a type of artificial neural network that can capture complex relationships in the data, leading to accurate predictions.

While Logistic Regression is a linear model and may not capture complex relationships as effectively as ensemble or neural network models, its performance indicates that it can be just as effective and have balanced performance in predicting NBA success based on NCAA performance metrics.

5.3 Results for Feature Importance

When examining the top features across various models, "DPORPAG" (Defensive Points Over Replacement Per Adjusted Game) emerged as consistently significant, suggesting its crucial role in evaluating player performance. This metric measures a player's defensive impact relative to a replacement-level player, considering factors like blocks, steals, and defensive rebounds.

Additionally, "AdjOE" (Adjusted Offensive Efficiency) and "TReb" (Total Rebound Percentage) were also consistently highlighted as essential features. AdjOE estimates a team's offensive efficiency against an average Division I defense, while TReb indicates the percentage of available rebounds a player secured while on the court.

In summary, the features "DPORPAG," "AdjOE," and "TReb" are key metrics in assessing player performance, with DPORPAG specifically focusing on defensive contributions.

These metrics provide valuable insights into a player's defensive prowess, offensive efficiency, and rebounding abilities, aiding in comprehensive player evaluations.

5.4 Algorithms and Interpretability

The Local Interpretable Model-agnostic Explanations (LIME) technique, employed to interpret the machine learning models used in this study, obtained some interpretable explanations for individual predictions of complex models.

This analysis underscores the importance of model interpretability and how LIME can be a valuable tool in better understanding the decisions of machine learning models in different contexts.

5.4.1 Feature Influence in Determined Instances

Across all models, the number of games played (GP) consistently emerges as a significant factor. However, its effect varies, being negatively influential in MLP, Random Forest, and XGBoost and positively influential in SVM and Logistic Regression.

While minutes played percentage (Min_per) and usage rate (usg) show varying degrees of influence across models, their effects are inconsistent. For example, Min_per has a consistently negative impact in MLP and Logistic Regression but is not as influential in SVM, Random Forest, and XGBoost. Similarly, usg harms MLP, SVM, and Random Forest, but its influence is positive in Logistic Regression and XGBoost. Effective field goal percentage (eFG) also exhibits divergent impacts, which are negatively influential in SVM and Logistic Regression but less so in other models.

5.4.2 Comparison Across Models

Each model provides unique insights into player prediction. For example, SVM places high importance on GP and Min_per, suggesting that consistent game participation and playing time are crucial for predicting NBA success.

On the other hand, Logistic Regression emphasizes GP positively but considers Min_per as a negative factor, indicating a nuanced view of player performance factors.

These differences in feature importance highlight the need for a nuanced approach to player evaluation. Decision-makers should consider each model's specific context and priorities when using them to inform player selection decisions.

6. CONCLUSION AND FUTURE WORK

Our Machine Learning-based Approach has yielded insightful outcomes and contributed to predicting NCAA basketball players' success in reaching the NBA [20]. The genetic algorithm successfully reduced the feature set from 65 to 28, showcasing its efficacy in feature selection. The selected attributes encompass various player statistics, highlighting key performance indicators for the NBA's success.

Random Forest and Multi-Layer Perceptron (MLP) emerged as top performers when evaluating the predictive models, achieving high accuracy rates. These models demonstrated strong classification abilities, particularly in identifying players with a high likelihood of NBA success while minimizing false positives. However, Logistic Regression, support vector machine (SVM), and XGBoost also performed well.

However, choosing the most suitable model should consider not only performance or generalization but also the interpretability aspect, which is even more valuable for selecting the best strategy for the problem, considering that the previous metrics were close.

The ensemble nature of Random Forest, where each tree contributes to a portion of the final decision, makes it difficult to understand how each variable influences the model's overall output. For MLPs, the weights and connections between the hidden layers are challenging to interpret, limiting an intuitive understanding of the predictions.

Evaluating this tradeoff, the Logistic Regression model has an advantage due to its simple structure and ability to produce more intuitive predictions through coefficients.

An interesting observation is the consistency in feature importance across different models. The features "dporpag," "adjoe," and "treb" emerged as consistently important in several models, indicating their significant influence on the prediction outcome. This underscores the importance of these player characteristics in determining a player's success in reaching the NBA.

In conclusion, applying the Local Interpretable Model-agnostic Explanations (LIME) technique has provided valuable insights into the decision-making processes of machine learning models in predicting the success of NCAA basketball players in reaching the NBA. The analysis revealed that while certain features such as the number of games played (GP) consistently play a significant role across models, the interpretation of other features like minutes played percentage (Min_per), usage rate (usg), and effective field goal percentage (eFG) varies widely among different models. This underscores the importance of model interpretability in understanding and utilizing machine learning models effectively, especially in complex and high-stakes decision-making scenarios such as player selection for professional sports leagues.

Furthermore, the comparative analysis of the models highlights the need for a nuanced approach to player evaluation, considering each model's specific context and priorities. For instance, while SVM emphasizes the importance of consistent game participation and playing time, Logistic Regression offers a more nuanced view by considering the percentage of minutes as positive and negative influencers. These insights are crucial for decision-makers in the basketball industry, providing them with a deeper understanding of the critical factors driving player success and guiding them in making more informed and effective player selection decisions.

Overall, our study highlights the utility of machine learning in analyzing NCAA basketball player data and offers valuable insights for decision-makers in player selection and talent development. The models' interpretability and performance demonstrate the potential of such approaches in enhancing decision-making processes in sports analytics.

In our immediate future work, we aim to enhance the validation of our models by focusing on their high accuracy and explainability in the context of monitoring selected players. We seek to explore how the labels and player statistics for key features translate into success in a professional setting, such as the NBA. This effort will provide deeper insights into the predictive capabilities of our models and their practical application in real-world scenarios, ultimately contributing to advancing player selection and talent development strategies.

REFERENCES

[1] Breinman, L.; Friedman, J.; Stone, C.; R.A Olshen. Classification and Regression Trees. Taylor & Francis, EUA, 1984. 364 p.

[2] Quinlan, J. R.; C4.5: Programs for Machine Learning. Morgan Kaufmann, 1993.

[3] Hilbe, Joseph M. (2009). Logistic Regression Models. Chapman & Hall/CRC Press. ISBN 978-1-4200-7575-5.

[4] Cortes, C.; Vapnik, V.; Support-vector networks. Machine Learning, P. 273–297, 1995.

[5] Bishop C. M.; Neural Networks for Pattern Recognition. Oxford University Press, 1995.

[6] Breiman, L.; Random Forests , Journal Machine Learning, Vol. 45, 2001, P. 5-32.

[7] Friedman J.H (2001). “Greedy function approximation: a gradient boosting machine.” Annals of Statistics, pp. 1189–1232.

[8] Babatunde, O. H., Armstrong, L., Leng, J., and Diepeveen, D. (2014). A genetic algorithm-based feature selection. Albert, A. A., de Mingo Lopez, L. F., Allbright, K., and Gomez Blas, N. (2021). A hybrid machine learning model for predicting usa nba all-stars. Electronics, 11(1):97.

[9] Gunning, D., Vorm, E., Wang, J.Y., Turek, M., DARPA's Explainable AI (XAI) Program: A retrospective. Applied AI LettersVolume 2, Issue 4, 2021.

[10] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador Garcia, Sergio Gil-Lopez, Daniel Molina, Richard Benjamins, Raja Chatila, Francisco Herrera. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. Information Fusion, Volume 58, 2020, Pages 82-115, ISSN 1566-2535, https://doi.org/10.1016/j.inffus.2019.12.012.

[11] Houde, M. (2021). Predicting the outcome of NBA games.

[12] Mahmood, Z., Daud, A., and Abbasi, R. A. (2021). Using machine learning techniques for rising star prediction in basketball. Knowledge-Based Systems, 211:106506.

[13] Albert, A. A., de Mingo Lopez, L. F., Allbright, K., and Gomez Blas, N. (2021). A hybrid machine learning model for predicting usa nba all-stars. Electronics, 11(1):97.

[14] Hsu, P.-H., Galsanbadam, S., Yang, J.-S., and Yang, C.-Y. (2018). Evaluating machine learning varieties for nba players’ winning contribution. In 2018 International Conference on System Science and Engineering (ICSSE), pages 1–6.

[15] Blum, A. L. and Langley, P. (1997). Selection of relevant features and examples in machine learning. Artificial intelligence, 97(1-2):245–271.

[16] Liashchynskyi, P.; Liashchynskyi, P. (2019) Grid Search, Random Search, Genetic Algorithm: A Big Comparison for NAS. arXiv preprint arXiv:1912.06059, [S.l], 2019.

[17] Zhou, Z-H. (2021). Machine learning. Springer Nature. https://doi.org/10.1007/978-981-15-1967-3.

[18] Test Setup using Google Computer Engine. (n.d.). Google Colab. Retrieved May 3, 2024, from https://colab.research.google.com/?hl=pt.

[19] NCAA Matches Dataset. (n.d.). GitHub. Retrieved May 5, 2024, from https://raw.githubusercontent.com/RubensBritto/data_NBA_Draft/main/Estatisticas%20Avancadas/CollegeBasketballPlayers2009-2021.csv

[20] Costa, Dante ; Fechine, Joseana ; Brito, José; Ferro, João ; Costa, Evandro ; Lopes, Roberta. A Machine Learning Approach Using Interpretable Models for Predicting Success of NCAA Basketball Players to Reach NBA. In: 16th International Conference on Agents and Artificial Intelligence, 2024, Rome. Proceedings of the 16th International Conference on Agents and Artificial Intelligence, 2024. v. 3. p. 758-765.

[21] Machine Learning study with NCAA data. (n.d.). Google Colab. Retrieved May 5, 2024, from https://colab.research.google.com/drive/1UAVXEK-Nk1oDgdYzBuuDYdwO92ng1aQB?authuser=1#scrollTo=5lKLP9aT3PyS