EECS 4404/5327 Project

In the constantly evolving world of competitive basketball, gaining a strategic edge is crucial for every team. As data becomes more readily available, the demand for advanced analytics in basketball strategy and the limitations of manual player assessment become more evident. Our project addresses the challenge of predicting basketball player roles. To do this, we explore various machine learning models and compare their performances.

The intricacies of our design involve harnessing the power of different models with features and other parameters specifically tailored for positional role prediction. The design processes historical player statistics and game data and extracts the most informative features as the primary inputs for the predictive models. The classifier models we decided on were Neural Networks, DecisionTree, RandomForest, ExtraTrees, KNeighbors, XGBoost and Logistic regression. To evaluate the models, we used several metrics such as accuracy, confusion matrices, precision and recall.

Our results showcase varying degrees of accuracy in predicting basketball player roles based on historical data. Despite our best efforts in trying to resolve the overfitting that the results imply, our best model achieves comparable accuracy to similar applications. As we move forward, we aim to further refine our techniques and continually enhance the accuracy and applicability of our models for the ever-evolving landscape of pro basketball.

Introduction

This application is a pioneering solution for data-driven decision-making in basketball, utilizing machine learning to predict player roles. With Sklearn for feature selection and model evaluation, alongside Pandas and NumPy for effective data handling, the application revolutionizes basketball strategy. By automating player role predictions, it empowers teams to optimize player selection, refine game strategies, and elevate overall team performance.

Basketball teams currently rely on subjective assessments and manual scouting, lacking detailed statistical insights into player roles. This application addresses these challenges by providing objective, data-driven predictions, reducing bias, and enhancing efficiency in player recruitment, game strategy optimization, and performance improvement.

The project is specifically tailored for the multiclass classification of male players in the NBA, utilizing historical player statistics and game data for training and testing. The scope of the model is refined to predict player roles, namely point guard (PG), shooting guard (SG), center (C), power forward (PF), and small forward (SF). To enhance data quality, rows with less than a 20-minute average playtime per game were intentionally filtered out. This meticulous data curation aims to reduce noise from outlier

performances, emphasizing the importance of meaningful player contributions within the dataset and aligning with the project's multiclass classification objectives.

Basketball teams rely on manual scouting and subjective evaluations to figure out what the player's roles and positions will be. There might be some statistics involved, but it's not in large detail. This can be a problem since teams can't make decisions that will also help them invoke better game strategies and development, either because of human bias or the subjectivity in the assessments. This application will provide a data-driven advantage to help recruit players that better form an improved strategy and performance, offering objective insights and making the scouting process more efficient.

Related Work

Similar applications attempting to predict basketball player roles exist, but there is no commercial product available. What sets our project apart is the emphasis on enhancing existing methodologies through the incorporation of feature selection techniques, regularization, normalization, and preprocessing. In contrast to the similar application we used to compare our model [6], our design specifically focuses on determining the optimal number of features while working with a larger dataset. Also, we opted for using different classifiers such as a Neural Network, XGBoost and Logistic Regression. This comprehensive approach aims to significantly improve the efficiency and accuracy of basketball role predictions, making our project distinct in the field.

Adjustments

The number of classes considered in the original project proposal was reduced. The original plan called for the possibility of expanding to include multiple positions—custom roles and positional roles. However, due to difficulties in obtaining a high-quality dataset for these additional classes within the time frame specified, they were removed from the final project scope. The emphasis shifted to optimizing predictions for the primary basketball player roles (PG, SG, C, PF, SF) without taking multiple roles into account.

Methodology

Our basketball player role prediction design pipeline follows a systematic approach, starting with data preprocessing where historical player statistics and game data are refined by handling missing values,

converting categorical data, and ensuring data integrity. Feature selection plays a crucial role, employing techniques like SelectKBest and ExtraTreesClassifier to identify pertinent features for model training. The models then take these selected features as input to predict player roles, including positions like Point Guard and Center. Evaluation metrics, including accuracy and confusion matrix, assess the model's performance, and iterative refinement, along with hyperparameter tuning, optimizes the overall predictive capability. This modular pipeline aims to enhance decision-making for basketball teams by providing data-driven insights into player selection and game strategy optimization.

Dataset

For the basketball player role prediction project, the dataset is sourced from historical player statistics and game data, forming a comprehensive repository with 31 columns representing various player attributes and performance metrics. This raw dataset undergoes essential pre-processing steps to ensure optimal model training and generalization. The preprocessing involves handling missing values, converting categorical data using sklearn's one-hot encoding, normalizing numerical features with MaxMinScaler to preserve the range between data points, and eliminating duplicates to maintain data integrity. Furthermore, a strategic decision is made to filter out rows with less than a 20-minute average playtime per game, contributing to noise reduction from outlier performances. This meticulous pre-processing enhances the quality and relevance of the dataset for training and testing the machine learning model, aligning with the specific requirements of the basketball player role prediction task.

Model Training

The model training was done for a different set of classifiers and a neural network to compare which one of them is better or more accurate.

The neural network was chosen for its ability to capture complex relationships in data and learn hierarchical representations. The non-linearity introduced by ReLU and other activation functions enhances the model's capacity to recognize intricate patterns in player statistics. The features from the feature selection process are taken as input and the output will be predicted basketball player roles that best fit the data. For the training process, the hidden layers are configured using ReLU activation functions and the output layer uses the Softmax activation function. We use the Adam optimizer for the purpose of extending stochastic gradient descent using an adaptive learning rate and quicker minimization of categorical cross-entropy loss. The training is done iteratively with multiple epochs and hyperparameter tuning is involved for adjusting different values so that a more optimal model can be found.

The classifiers chosen are decision tree, random forest, extra trees, XGBoost, and K-nearest neighbours. Decision trees and ensemble methods are known for their interpretability and versatility in classification problems, and K-nearest neighbours provides a simple yet effective classification approach. The input takes in selected features and each model is trained under different circumstances to output the predicted player roles. The training process between each of the models is similar in terms of fitting the model to the training data and decision trees as well as ensemble methods are trained to optimize entropy gain or Gini impurity, while K-nearest neighbours is trained to classify instances based on the majority class of their neighbours.

With respect to the first and second parts of working on this project, no changes were needed since the goal was clearly defined and achievable as stated in the first part and looking through different studies in the second part only influenced how we should evaluate our model and which performance metrics to use, nothing with respect to training or using a different model architecture.

Prediction

The model generates predictions for the player's basketball role based on the learned patterns. The prediction indicates whether the player is best suited for positions like shooting guard (SG), power forward (PF), center (C), small forward (SF), or point guard (PG).

Through many iterations of backpropagation, the neural network will be able to find a particular set of weights and biases that minimize the cross-entropy cost in determining positions. Predicting with new data involves reshaping/scaling the input, and feeding it forward through the network where the learned weights and biases process the data points at each layer and output the leading result.

Decision trees and ensemble methods should hold the most significant features at the end of training, then the new data points are passed through the branches and the best or majority position is outputted at the end of the process.

Based on the proximity of training data, the K-Nearest-Neighbours classifier will have laid out groups for certain positions based on distances and it finds patterns in the new data by considering the majority classification of the closest points from training.

Performance Evaluation

- Task 1: Classification of basketball player roles on the training set.

- Task 2: Generalization assessment on the testing set.

To evaluate our design, we set aside 18% of the dataset for testing. This data has not been used for training, so it simulates new data that can be used to test how well the model can predict the role of a player given never-before-seen data.

From the testing data, we computed several metrics to evaluate the models such as accuracy, a confusion matrix, precision, recall, and f-scores. From these results, we determine the best-performing model for our design and compare it to a similar application [6] as the baseline.

Results

Decision Tree Classifier

Training set accuracy: 98%

Testing set accuracy: 55%

Random Forest Classifier

Training set accuracy: 99%

Testing set accuracy: 53%

Extra Trees Classifier

Training set accuracy: 99%

Testing set accuracy: 51%

K-Neighbors Classifier

Training set accuracy: 98%

Testing set accuracy: 53%

XGBoost Classifier

Training set accuracy: 99%

Testing set accuracy: 65%

- This classifier gave us the best result, generalizing well to new data.

Neural Network

Training set accuracy: 59%

Testing set accuracy: 57%

Logistic Regression

Training set accuracy: 65%

Testing set accuracy: 67.5%

XGBoost Test Classification Report:

	precision	recall	f1-score	support
0	0.70	0.65	0.67	256
1	0.57	0.59	0.58	286
2	0.78	0.82	0.80	300
3	0.64	0.61	0.63	297
4	0.65	0.66	0.66	313
accuracy			0.67	1452
macro avg	0.67	0.67	0.67	1452
weighted avg	0.67	0.67	0.67	1452

XGBoost Training Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	2234
1	1.00	1.00	1.00	2635
2	1.00	1.00	1.00	2685
3	0.99	0.99	0.99	2625
4	0.99	1.00	0.99	2885
accuracy			1.00	13064
macro avg	1.00	1.00	1.00	13064
weighted avg	1.00	1.00	1.00	13064

[LibSVM]Classification Report (Polynomial Kernel):

	precision	recall	f1-score	support
0	0.72	0.56	0.63	256
1	0.55	0.62	0.58	286
2	0.77	0.84	0.81	300
3	0.67	0.61	0.64	297
4	0.67	0.71	0.69	313
accuracy			0.67	1452
macro avg	0.68	0.67	0.67	1452
weighted avg	0.68	0.67	0.67	1452

Classification Report (RBF Kernel):

	precision	recall	f1-score	support
0	0.71	0.51	0.59	256
1	0.52	0.58	0.55	286
2	0.80	0.76	0.78	300
3	0.61	0.57	0.59	297
4	0.61	0.73	0.66	313
accuracy			0.64	1452
macro avg	0.65	0.63	0.63	1452
weighted avg	0.65	0.64	0.64	1452

Discussion

The outcomes of our basketball player role prediction project prompt a re-evaluation of our initial expectation to create a dominant neural network. While we aspired for superior performance, the results show the complexity of the task. Exploring various models, including alternative classifiers, revealed nuanced challenges in predicting basketball player roles. Despite intensive efforts, the observed insignificance in the improvement of the neural network highlights the need for a deeper understanding of the underlying dynamics to enhance model efficacy. This insight necessitates a shift in perspective from achieving immediate dominance to a more intricate understanding of the unique challenges posed by basketball player role prediction.

Also, it is clear from the discrepancy in training and testing set accuracy that all our models except for Neural Network are overfitting heavily to the training set. It might be worthwhile to pursue the root cause of this despite efforts to mitigate overfitting with early stopping and experimenting with fewer features.

One of the strengths of our design lies in the meticulous feature selection process, which involves a combination of techniques. By initially considering all available features from the dataset and subsequently employing SelectKBest, ExtraTreesClassifier and RFECV methods, we ensured the identification of the most relevant features for predicting basketball player roles. This strategic approach enhances the efficiency of our model by focusing on key attributes, contributing to improved interpretability and potentially reducing overfitting.

Additionally, our design exhibits strength in the preprocessing phase, where careful data handling techniques were applied. The conversion of categorical data, such as player positions, into numerical representations using sklearn's one-hot encoding, contributes to the model's ability to comprehend these essential attributes. The normalization step, achieved through MaxMinScaler, ensures that the numerical values fall within a standardized range, preventing biases and allowing for consistent comparisons between different features. These strengths in feature selection and preprocessing collectively contribute to the robustness and effectiveness of our design for basketball player role prediction.

The design for predicting basketball player roles exhibits notable limitations. It relies on historical player statistics and game data, potentially missing real-time changes in player performance and strategic dynamics. The exclusion of features like real-time player performance data and player heights may hinder the model's adaptability to evolving scenarios. Focusing exclusively on male players in the NBA limits the generalizability, requiring adjustments for the inclusion of female players or those from different leagues. The dataset's quality and representativeness can influence model performance, emphasizing the importance of diverse data sources. Additionally, the exclusion of players who play multiple roles may impact accuracy in scenarios where players exhibit versatile skills across different positions.

Future Directions

A key objective for future development is to extend the current basketball player role prediction model into a more sophisticated multi-label classifier. This enhancement aims to accurately categorize players into multiple roles within a single prediction, offering a more nuanced and insightful understanding of player capabilities. The improved model would contribute to streamlined decision-making for basketball teams, eliminating the need for individual predictions for each player’s role. During the initial

development, challenges were encountered in procuring a dataset that comprehensively covers the diversity of player roles. Future efforts will be directed towards creating a custom dataset or utilizing open-source pre-trained models tailored to basketball player role prediction.

Additionally, another crucial direction for future development involves expanding the model's scope to encompass a broader range of features and attributes relevant to basketball player roles. This improvement could be achieved by incorporating additional datasets or leveraging advanced classification features from existing pre-trained models. By enhancing the model's learning capabilities, it becomes more versatile and effective in capturing the complexities of various player roles, ultimately improving its utility and impact in diverse basketball scenarios.

References

1. DataSet
2. Source Code used from other applications
3. XGBoost technique
4. Suggestions used for Neural Network
5. Information used for Feature Selection
6. The similar application for comparison
7. Information used for Feature Scaling