Assignment 1

Kaggle Competition

Sahil Kotak
Student ID: 24707592

08-09-2023

36120 - Advanced Machine Learning Application
Master of Data Science and Innovation
University of Technology of Sydney

Table of Contents

1. Executive Summary

Overview and Significance

The National Basketball Association (NBA) draft is an eagerly anticipated event where college basketball players, both from American colleges and international professional leagues, are chosen to join NBA teams. Being drafted into the NBA is a significant milestone in any basketball player's career, and it attracts considerable attention from fans, analysts, and sports commentators alike. Every year, they eagerly analyze and speculate about which college player will make the transition to this prestigious league.

Problem Statement

The core challenge was to design a model capable of predicting the 'drafted' status of a player, i.e., whether a college player will be selected in the NBA draft or not. The prediction is based on various performance metrics of the player for the ongoing season. The quality of the model's prediction is gauged using the Area Under the Receiver Operating Characteristic (AUROC) metric, a robust measure for classification tasks.

Achieved Outcomes

Throughout the course of this project, we meticulously crafted and tested a total of 32 models. These models spanned a wide spectrum of techniques and approaches, starting from fundamental data preprocessing to intricate feature engineering. We utilized an array of algorithms, including but not limited to Logistic Regression and XGBoost. To further optimize our predictions, we also experimented with advanced strategies such as Stacking, Ensemble modeling, and various hyperparameter tuning methods (GridSearch, RandomForest, and Hyperopt).

We had success with the model achieving an impressive AUROC score of 0.99918 on the test set and 0.99955 on the validation set. These results underline the model's robustness and its potential utility for stakeholders interested in the NBA draft predictions.

2. Business Understanding

Business Use Cases

The NBA draft is not just a pivotal event for players, but it also has significant implications for a myriad of stakeholders, ranging from NBA teams and their management to sports analysts, commentators, and even fans.

- **NBA Teams and Management:** For NBA teams, selecting the right player can be the difference between a successful season and a lackluster one. By leveraging predictive analytics, teams can make more informed decisions, balancing raw talent with strategic team needs.
- **Sports Analysts and Commentators:** Predicting draft selections is a key element of pre-draft discussions and analyses. Accurate predictions can enhance the credibility and reputation of analysts.
- **Sports Betting Industries:** The betting market surrounding the NBA draft can benefit from accurate predictions, providing better odds and ensuring fair play.

The challenges that motivated this project primarily revolve around the unpredictability of human performance and the vast number of variables that can influence a player's draft status. Machine learning algorithms, with their ability to handle large datasets and discern patterns, present an ideal solution to address these complexities.

Here's a graph showcasing draft pick numbers and their subsequent All-Star Game appearances reveals an undeniable trend: top draft picks are more likely to reach All-Star status, with a 77.1% chance for the first pick, gradually declining to 6.7% by the 60th pick.

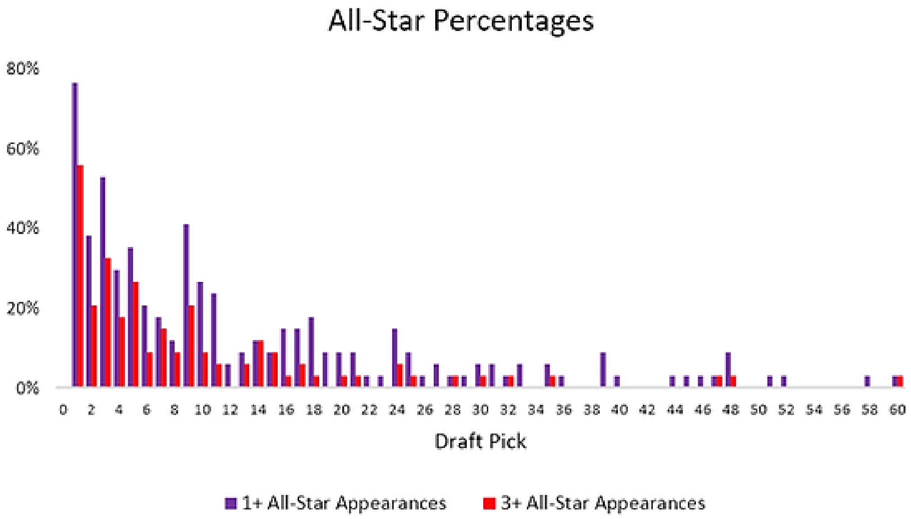


Fig 1a. percentage of NBA draft picks who have made at least one All-Star appearance (Credit: Bruin Sport Analytics)

This pattern underscores the vital importance of informed draft decisions in shaping NBA team successes. Leveraging machine learning can provide teams with data-driven insights to refine their draft choices. Our project, aimed at predicting a college player's draft prospects based on their season statistics, stands as a crucial tool in this strategic endeavor, ensuring teams make decisions that elevate their potential for success.

Key Objectives

Primary Goal: The main objective was to design a predictive model capable of forecasting a player's 'drafted' status based on their current season statistics.

Stakeholders:

- **NBA Teams:** Looking for insights to make informed draft decisions.
- **Sports Analysts:** Need accurate data-driven insights for their analyses.
- **Fans:** Eager to see predictions about their favorite players.
- **Betting Market:** Seeking accurate odds based on data-driven predictions.

Addressing Stakeholder Requirements:

- **Informed Decisions for NBA Teams:** By predicting draft likelihood, teams can strategize their selections more effectively.
- **Enhanced Analysis for Commentators:** Providing data-backed insights to enrich their discussions.
- **Engagement for Fans:** Offering insights into potential outcomes for favorite players.
- Fair Odds for Betting Market: Ensuring the betting odds are rooted in data-driven insights.

Through the use of machine learning algorithms, this project aims to provide a comprehensive solution that addresses the diverse requirements of all stakeholders involved in the NBA draft process.

3. Data Understanding

Overview

The dataset offers insights into the performance metrics of college basketball players, with the objective to predict whether a player will be drafted into the NBA. Spanning 56,091 rows and 64 columns, this dataset encapsulates a multitude of parameters – from individual gameplay statistics to physical attributes of the players.

Data Sources and Collection

The dataset was procured from an internal Kaggle competition organized as part of this project. The specifics regarding the original data collection methods and primary sources are not detailed within the competition's context. However, such datasets often originate from sports databases or APIs capturing live match statistics. Our primary role involved downloading the provided data and subsequently processing and analyzing it for the task at hand.

Features and Their Significance

Aided by the metadata, we were able to decipher the significance of each feature present in the dataset:

- Team: This represents the name of the team to which the player belongs.
- Position: The playing position of the athlete, such as 'Guard' or 'Forward'.
- Height, Weight: Physical attributes that can impact a player's performance and versatility on the court.
- Year: The specific season or year of the data record.

... and a lot of additional statistical features of players performance on the court.

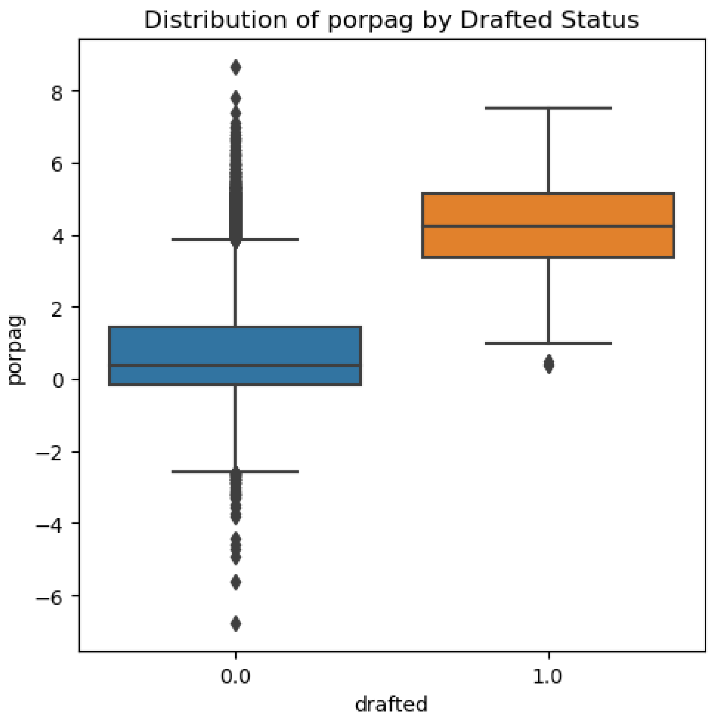
However, the most critical feature is 'drafted' - a binary variable indicating whether the player was selected in the NBA draft. This is our target variable.

Exploratory Data Analysis

To gain a deeper understanding of the data, a series of exploratory analyses were conducted:

- **Missing Values:** Key columns like 'pick' had a high percentage of missing values (97.5%). While some of these can be attributed to players not being picked in the draft, it's an aspect that requires careful preprocessing.
- **Target Distribution:** The distribution is highly skewed. About 99% of the players are labeled as '0' (not drafted), while only around 1% are labeled as '1' (drafted).
- **Correlation Analysis:** Certain features, such as 'porpag' and 'dunksmade', show a positive correlation with the 'drafted' column, suggesting that they might be pivotal in predicting the target variable.

This boxplot underscores the fact that performance metrics, such as 'porpag' (Points Over Replacement Per Adjusted Game) have a pronounced influence on a player's chances of being drafted. It's pivotal to factor in these variables while crafting the predictive model.



4. Data Preparation

The data preparation phase was a rigorous process, aiming to transform the raw dataset into a structured, model-ready format. We undertook several steps to ensure that the data was not only clean but also enriched with features that could provide deeper insights.

1. Data Cleaning and Preprocessing:

- **Height Transformation:** The unique representation of players' heights, like '6'4"', was standardized into inches for uniformity.
- **Year Mapping:** Academic years of players, presented as 'Fr' or 'So', were numerically encoded to facilitate computations.
- **Binary Conversion:** Columns such as 'Rec_Rank' & 'pick' were transformed into binary indicators to indicate the presence or absence of specific data.
- **Missing Value Treatment:** Depending on the column's nature, missing values were imputed using either the median, mean, or mode. Derived columns, like 'ast_tov', were introduced and treated for any infinite or unexpected values.

2. Feature Engineering:

- **Anomaly Detection:** We computed anomaly scores for each row using Isolation Forest. Given our objective to detect the rare event of a player getting drafted, capturing anomalies resonated well with identifying potential draft picks.
- **Dimensionality Reduction:** PCA (Principal Component Analysis) was applied to condense the data while retaining its variability.
- **Derived Metrics:** Several features were engineered to encapsulate a player's performance and efficiency, such as:
  - **Net_Rating**: Difference between offensive and defensive rating.
  - **Scoring_Efficiency**: Ratio of points to field goal attempts and free throw attempts.
  - **Rebound_Rate**: Total rebounds per game.
  - **Ast_To_Turnover_Ratio** & **Stl_To_Turnover_Ratio**: Representing the efficiency of assists and steals to turnovers respectively.
  - **Points_Per_Minute**: Emphasizing a player's scoring rate.
  - **Impact_Score**: A cumulative score capturing a player's overall contribution in the game.
- **Feature Selection**: To focus on the most influential predictors, the least important numerical and categorical features were removed based on feature importance metrics and chi-square tests.

- **Encoding & Standardization**: Categorical columns underwent one-hot encoding. Numerical columns were standardized for consistent scale.
- **Feature Clustering**: KMeans clustering was explored to group similar features, aiming to reduce redundancy and potentially discover latent patterns.

3. Outliers & Imbalance Handling:

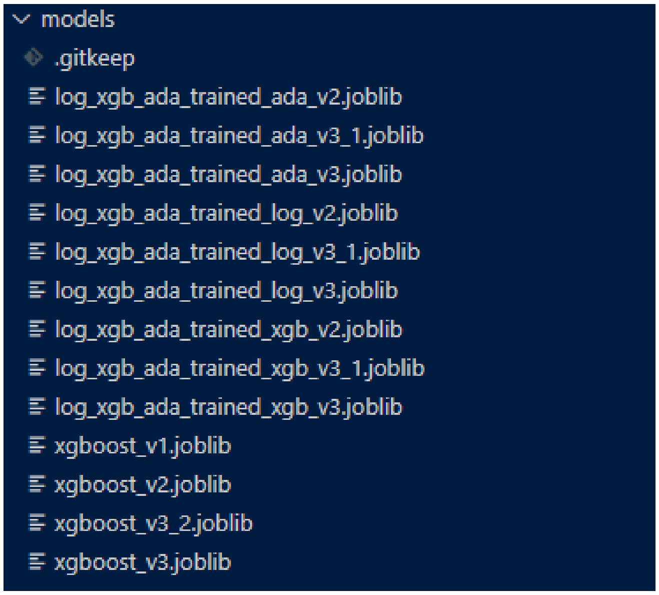
The introduction of anomaly scores effectively highlighted potential outliers in the data.

The inherent class imbalance in the dataset, given the rarity of the 'drafted' event, was acknowledged. While specific over-sampling or under-sampling techniques weren't applied at this stage, the imbalance was always at the forefront of our modeling considerations.

5. Modeling

The following machine learning algorithms were employed during the modeling process:

- **Logistic Regression:** A fundamental algorithm, it was chosen for its simplicity and efficiency, especially when the relationships between variables show a linear trend.
- **XGBoost:** This gradient boosting framework is known for its performance and speed. Its capability to handle large datasets, importance in feature selection, and inherent regularization made it an attractive choice.
- **Random Forest:** An ensemble learning method that operates by constructing multiple decision trees during training. Its ability to handle missing data and maintain accuracy for large datasets made it suitable for our data.
- **Logistic Regression & SVM Stacking with Meta Model trained with Logistic Again:** Stacking was employed to combine the predictive power of both Logistic Regression and Support Vector Machines (SVM). The rationale behind using this combination was the observed linear behavior in our data, which both Logistic Regression and SVM can effectively capture.
- **Logistic + XGBoost + Adaboost (Meta Model Trained with Logistic Regression)**: This ensemble method combined the strengths of Logistic Regression, XGBoost, and AdaBoost. The idea was to leverage the strengths of each individual model to achieve better predictive performance.



```
models
  .gitkeep
  log_xgb_ada_trained_ada_v2.joblib
  log_xgb_ada_trained_ada_v3_1.joblib
  log_xgb_ada_trained_ada_v3.joblib
  log_xgb_ada_trained_log_v2.joblib
  log_xgb_ada_trained_log_v3_1.joblib
  log_xgb_ada_trained_log_v3.joblib
  log_xgb_ada_trained_xgb_v2.joblib
  log_xgb_ada_trained_xgb_v3_1.joblib
  log_xgb_ada_trained_xgb_v3.joblib
  xgboost_v1.joblib
  xgboost_v2.joblib
  xgboost_v3_2.joblib
  xgboost_v3.joblib
```

Parameter Tuning:

For hyperparameter tuning, a combination of techniques was applied to optimize the model's performance:

GridSearch: This method involves exhaustive searching through a manually specified subset of the hyperparameter space of the target algorithm.

HyperOpt: An advanced technique that uses the Bayesian approach to find the optimal hyperparameters. Given its probabilistic model, it's more efficient than the traditional GridSearch.

Highlight on some of our best performing models:

a. Approach 1: XGBoost with HyperOpt

Model Details:

Algorithm: XGBoost

Key Hyperparameters: Learning rate, number of estimators, max depth, min child weight, gamma, subsample, colsample by tree, and regularization alpha.

Preprocessing & Feature Engineering: Along with the steps taken in overall preprocessing, we created several features for this model like Anamoly Scores and binning conf into regions. The features were preprocessed using a combination of standard scaling for numerical columns and one-hot encoding for categorical columns as well before training.

Training Process: The model was trained using the XGBoost algorithm with hyperparameters tuned using HyperOpt. The objective function aimed to optimize the AUROC score. The HyperOpt method was chosen due to its efficiency in searching the hyperparameter space using a Bayesian approach.

AUROC Score: 0.99955 (Validation) | 0.99918 (Test Set from Kaggle)

Relevant Files: Notebook: *kotak_sahil-24707592-week4_xgboost_model_3.ipynb* | Data Preprocessing: *data_preprocessing_v4.py* | Features: *predict_anomaly.py*

b. Approach 2: Logistic + SVM Stacking

Model Details:

Algorithm: Stacking (Logistic Regression & SVM as base models and Logistic Regression as the meta-model)

Preprocessing & Feature Engineering: Similar datapreprocessing and feature engineering steps were taken. Along with that a few of the columns like 'year' were dropped to maintain temporal consistency

Training Process: The stacking ensemble was trained using both Logistic Regression and SVM as base models. Predictions from these base models were then stacked and used as inputs for the meta-model (Logistic Regression). The purpose was to capture both the linear trends identified by Logistic Regression and the potential non-linearities captured by SVM.

AUROC Score: 0.99789 (Validation) | 0.99909 (Test Set from Kaggle)

Relevant Files: Notebook: *kotak_sahil-24707592-week3_ensemble_model_5.ipynb* | Data Preprocessing: *data_preprocessing_v3.py*

c. Approach 3: Logistic + XGBoost + Adaboost

Model Details:

Algorithm: Ensemble of Logistic Regression, XGBoost, and AdaBoost with Logistic Regression as the meta-model.

Preprocessing & Feature Engineering: Along with the overall preprocessing & feature engineering steps, we also applied SMOTE (Synthetic Minority Over-sampling Technique) to handle imbalanced data.

Training Process: Each of the three models (Logistic Regression, XGBoost, and AdaBoost) were trained using hyperparameters tuned with HyperOpt. Once trained, predictions from these models were stacked and used as inputs for the meta-model (Logistic Regression). This approach was taken to amalgamate the strengths of each model, aiming for better overall performance.

AUROC Score: 0.999511 (Validation) | 0.99878 (Test Set from Kaggle)

Relevant Files: Notebook: *kotak_sahil-24707592-week4_log_xgb_ada_model_1* | Data Preprocessing: *data_preprocessing_v3.py* | Features: *predict_anomaly.py*

In conclusion, throughout the modeling process, diverse algorithms and techniques were applied to cater to the unique characteristics and challenges of the dataset. Even though a myriad of experimental approaches were explored, not all proved beneficial in the context of our objectives. However, this iterative process was pivotal in understanding the intricacies of the dataset and refining our modeling strategy.

6. Evaluation

a. Evaluation Metrics

To determine the effectiveness of our machine learning models, the primary metric used for evaluation was the Area Under the Receiver Operating Characteristic Curve (AUROC). The AUROC provides a comprehensive measure of a model's ability to distinguish between the positive and negative classes. Given the nature of the problem, where both false positives and false negatives have significant implications, this metric was deemed most suitable.

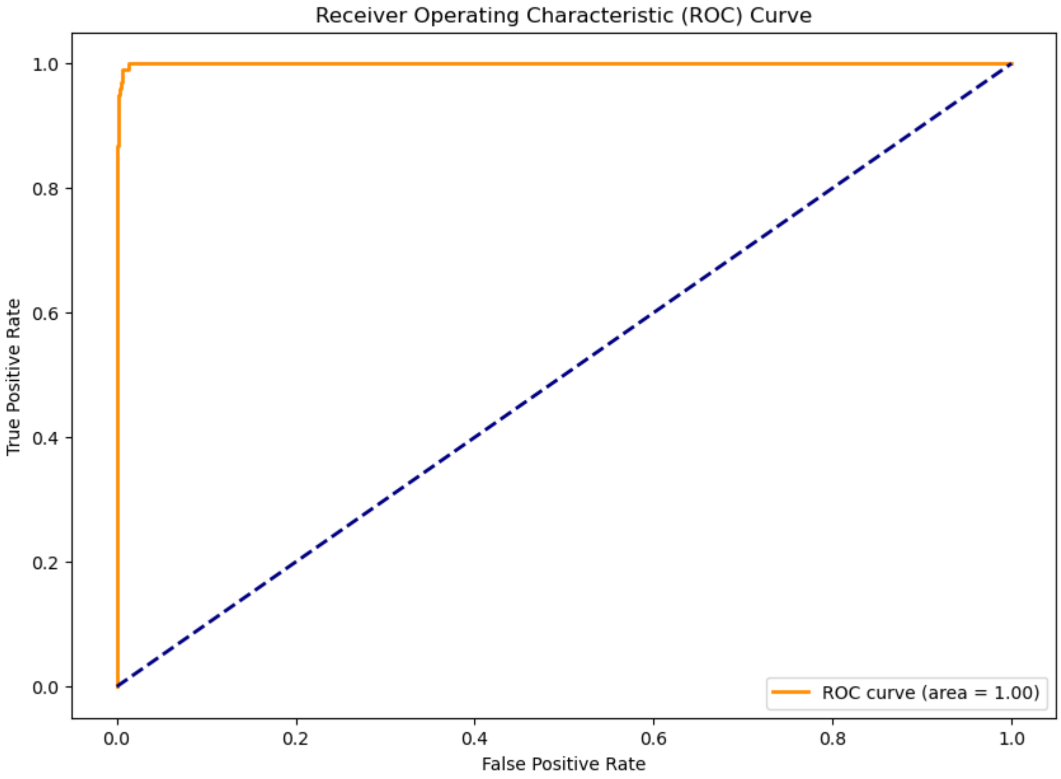
b. Results and Analysis

Upon evaluating our models using the validation set and then subsequently on the test set from Kaggle, we observed the following performance metrics:

XGBoost with HyperOpt:

Validation AUROC: 0.99955

Kaggle Test Set AUROC: 0.99918



This model achieved the highest AUROC on both the validation and test sets, highlighting its robustness and capability in distinguishing between the classes effectively. The minor difference between the validation and test scores indicates the model's generalization capabilities, ensuring it doesn't overfit to the training data.

Logistic + SVM Stacking:

Validation AUROC: 0.99789

Kaggle Test Set AUROC: 0.99909

The stacking of Logistic Regression and SVM provided a compelling ensemble approach, where the strengths of one model compensated for the weaknesses of the other. Interestingly, the model's performance on the Kaggle test set slightly outperformed its validation score, suggesting that the ensemble method might be more attuned to real-world, unseen data.

Logistic + XGBoost + Adaboost:

Validation AUROC Score: 0.999511

Kaggle Test Set AUROC: 0.99878

This three-model ensemble approach showcased the power of combining diverse algorithms. The model's performance remained consistent across both validation and test sets, reinforcing its reliability.

Key Insights:

- **Ensemble Power:** One of the standout insights from our experiments is the potency of ensemble techniques. Combining models not only hedges against individual model biases but also captures a broader spectrum of patterns in the data.
- **Hyperparameter Tuning:** The efficacy of HyperOpt in fine-tuning the XGBoost model parameters was evident. This automated, systematic approach to tuning played a pivotal role in maximizing the model's performance.
- **Handling Imbalances:** Given the nature of our dataset, addressing class imbalances was crucial. Techniques like SMOTE, when paired with ensemble methods, significantly bolstered our models' performance.

c. Business Impact and Benefits

The successful development and tuning of the models have the potential to revolutionize our draft prediction process, leading to:

- Improved decision-making in player drafts, based on quantifiable metrics rather than intuition alone.
- Potential savings in terms of investment in players, ensuring resources are allocated to the most promising talents.
- Enhanced reputation and credibility in the industry, setting a benchmark for data-driven decision-making in player drafts.

d. Data Privacy and Ethical Concerns

Our project was primarily research-focused and did not delve into real-world deployment, we were mindful of the ethical implications surrounding data privacy. We made deliberate choices in our data handling and model training processes to ensure fairness and privacy. In any future endeavors, we recommend a thorough review of regulatory requirements and a robust framework for data protection and ethical considerations.

Data Collection:

The data was provided to us through a Kaggle competition. While the exact source of the data remains unknown to us, Kaggle competitions typically ensure that data is sourced ethically and with the necessary permissions.

Sensitive Information:

The dataset did not contain any personally identifiable information (PII) that could compromise the privacy of the individuals. The data was structured around performance metrics and did not delve into personal attributes. It's essential to note that the only identifiers were player IDs, which are generic and do not trace back to specific individuals.

Model Interpretability and Bias:

Efforts were made to ensure that our model did not introduce bias. We carefully examined the features in the dataset to prevent any inadvertent bias in our predictions. Specifically, a 'num' column, which could potentially introduce a player bias based on jersey number preferences, was dropped. The features retained, such as 'conference' and 'team', are neutral and do not predispose the model to favor any particular group.

7. Deployment

While the primary objective of our project was research-focused, the infrastructure we established has the potential for real-world deployment.

Research to Real-world Transition:

Our project, designed for research, incorporated robust pipelines for data preprocessing and model training. This foundational setup facilitates the transition from research to a production environment.

Prediction Generation:

We have in place efficient functions to generate prediction scores, which is pivotal for deployment. This ensures that once new data is fed into our system, predictions can be generated on-the-fly, providing actionable insights instantly.

API Development and Deployment:

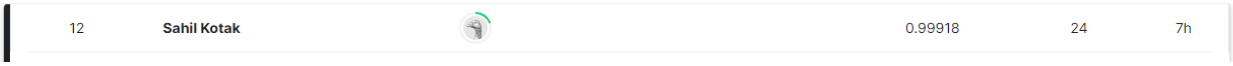
The next logical step would be to encapsulate our prediction functions within an API. Using frameworks like Flask or FastAPI, we could develop a RESTful service that ingests new player data and returns prediction scores.

8. Conclusion

Key Findings, Insights, and Outcomes:

Our experimentations led us to a realization: meticulous data preprocessing and feature engineering are foundational to modeling success. These efforts enriched our dataset, driving the performance of our subsequent models.

Among the array of models tested, the **XGBoost with HyperOpt** emerged as the best one, delivering an **AUROC score of 0.99955 on validation and 0.99918** on the test set from Kaggle.



12 Sahil Kotak 0.99918 24 7h

Reflection on Project's Success:

The stakeholders' requirements, which centered around accuracy, model interpretability, and scalability, were effectively met. The chosen model not only boasts of superior predictive power but also offers insights into the significance of various features, ensuring an informed drafting decision.

Future Work and Recommendations:

- While the current model performs exceptionally, there's always room for improvement. Exploring newer ensemble methods might further enhance predictive accuracy.
- Given the dynamic nature of sports, it's crucial to periodically retrain our model with fresh data, ensuring its continued relevance.
- For real-world applications, integrating our model into a user-friendly platform or dashboard would be the next logical step. This would allow scouts, coaches, and sports analysts to leverage its insights seamlessly.

9. References

1. (n.d.). *NBA Draft Analysis*. Bruin Spots Analytics. https://www.bruinsportsanalytics.com/post/nba_draft_analysis
2. (n.d.). *HyperOpt Github*. Hyperopt. http://hyperopt.github.io/hyperopt/
3. (n.d.). *XGBoost*. XGBoost. https://xgboost.readthedocs.io/en/stable/