EXPERIMENT REPORT

Student Name	Vishal Raj
Project Name	NBA Draft Prediction
Date	25th August 2023
Deliverables	Vishal_Raj-14227627-week2_RF2ndExp.ipynb Random Forest with feature selection Github Repository: https://github.com/vishalraj247/NBA_Draft_Prediction.git

1. EXPERIMENT BACKGROUND

1.a. Business Objective

The primary goal of this project is to predict whether a college basketball player will be drafted to join the NBA league based on his statistics for the current season. This prediction is of paramount importance for various stakeholders:

- **NBA Teams**: Accurate predictions can guide team managers and coaches in their draft decisions, helping them identify potential star players or hidden gems.
- **Sports Commentators and Analysts**: Having a data-driven model can provide commentators with insights to enrich their discussions and analyses during draft events.
- **Players and Their Coaches**: For college players aspiring to join the NBA, understanding their draft likelihood can help them focus on areas of improvement.
- **Fans**: NBA fans are always eager to know about the next big player. Predictions can enhance their engagement and discussions around draft events.

The impact of accurate predictions is multifaceted. It could mean the difference between securing a future star player or missing out on one for the NBA teams. For players, it could provide clarity on their career trajectory. However, incorrect predictions could lead to misinformed decisions, potential missed opportunities for teams, and misplaced expectations for players.

1.b. Hypothesis

Given the statistics of a college basketball player for a particular season, it is hypothesised that certain performance metrics can be indicative of a player's likelihood of being drafted into the NBA. Historically, players with higher points, assists, rebounds, and other key metrics have been observed to have a higher chance of being drafted. This hypothesis is grounded in the belief that NBA teams look for players who have demonstrated exceptional skills and performance during their college years. By analysing these statistics, we aim to uncover patterns and relationships that can predict a player's transition from college basketball to the NBA.

1.c. Experiment Objective

Again, this experiment aims to build a predictive model to determine the probability of a player being drafted accurately. The expected outcome is an AUROC score that is significantly higher than random guessing (0.5) and the previous experiment. By achieving this, we hope to provide a tool to assist various stakeholders, from sports analysts to NBA teams, in making informed decisions about potential draft picks. The possible scenarios resulting from this experiment include:

- **Highly Accurate Model**: This would validate our hypothesis and provide a reliable tool for predicting draft outcomes.
- **Moderately Accurate Model**: While imperfect, this would still offer valuable insights and could be improved with further experimentation.

Inaccurate Model: This would indicate that other factors, possibly outside of the provided statistics, play a significant role in draft decisions.

2. EXPERIMENT DETAILS

2.a. Data Preparation

Data preparation is crucial in any machine learning project, ensuring that the data fed into the model is clean, relevant, and structured appropriately. For this experiment:

- **Data Loading**: Multiple datasets were sourced, including ‘**train.csv’**, ‘**test.csv’**, ‘**sample_submission.csv’**, and ‘**metadata.csv’**. Each dataset provided unique insights into player statistics and draft outcomes.
- **Handling Missing Values**: The presence of missing values can skew predictions and reduce the reliability of a model. The ‘**pick’** column, which had a high percentage of missing values, was dropped to maintain data integrity. Other missing values were addressed using appropriate imputation techniques like using median and mode, ensuring no data point was left out due to incomplete information, now, with the help of defining a function and class.
- **Data Splitting**: The data was divided into training and validation sets to evaluate the model's performance. This ensures that the model is tested on unseen data, objectively assessing its predictive capabilities.

2.b. Feature Engineering

Feature engineering is a crucial step in the machine learning pipeline, ensuring that the data is in the best possible format for model training. In this experiment, several techniques were employed to enhance the predictive power of the data:

- **Handling Missing Values**: The ‘**DataPreprocessor’** class was introduced to manage missing data. Numeric columns with missing values were filled using their median, while non-numeric columns were filled using their mode. This ensures that no data point is discarded due to missing values, maximizing the information available for model training.

- **Feature Selection**: The ‘**Feature’** class was utilised to select important features based on their importance and inter-feature correlation. Features with an importance above a set threshold were retained. However, to avoid multicollinearity, features that were highly correlated with each other were removed. This ensures that the model is trained on relevant features without redundancy.

- **Class Imbalance**: The target variable, 'drafted', showed an imbalance in its distribution. To address this, the Synthetic Minority Over-sampling Technique (SMOTE) was employed. This technique generates synthetic samples in the feature space to balance out the classes, ensuring that the model is not biased towards the majority class.

- **Feature Scaling**: All features were scaled using the ‘**StandardScaler’** to ensure they have a mean of 0 and a standard deviation of 1. This is particularly important for algorithms that rely on distance metrics or gradient descent, ensuring that all features contribute equally to the model's predictions.

2.c. Modelling

The modeling phase is crucial as it determines the effectiveness of the predictions. The ‘**Model’** class was utilised here. Here's a breakdown of the steps taken in the updated notebook:

- **Model Selection**: The Random Forest Classifier remains the model of choice. It's an ensemble method that combines multiple decision trees to produce a more generalized model. This approach is known for its ability to handle large datasets with higher dimensionality and can identify complex nonlinear relationships.

- **Feature Selection**: Instead of manually selecting features based on domain knowledge, the updated notebook employs a more systematic approach. It uses the feature importances provided by the Random Forest model to select the most relevant features. This method ensures that only the most impactful features are used for training, which can lead to better model performance.

- **Hyperparameter Tuning**: The notebook employs ‘GridSearchCV’ for hyperparameter tuning. This method performs an exhaustive search over the specified parameter values for the Random Forest model. The main goal is to find the combination of parameters that provides the best performance. In the updated notebook, parameters such as **n_estimators**, **max_depth**, **min_samples_split**, **min_samples_leaf**, and **bootstrap** were tuned. The optimal values were identified based on the ROC-AUC score, ensuring that the model is both accurate and robust.

- **Model Training**: After selecting the best hyperparameters, the model is trained on the entire training dataset. This ensures that it captures all the patterns and relationships in the data, leading to better generalization on unseen data.

- **Validation**: The model's performance is validated using a separate validation set. This step is crucial to ensure that the model is not overfitting to the training data and can generalize well to new, unseen data.

By following these steps, the modelling phase ensures that the predictions are both accurate and reliable, making them valuable for any subsequent decision-making processes.

3. EXPERIMENT RESULTS

3.a. Technical Performance

Assessing the model's technical performance is pivotal to ensure its reliability and effectiveness. The ROC-AUC score, which gauges the model's capability to differentiate between the positive and negative classes, was the chosen metric:

- **Initial Model Performance**: Prior to any hyperparameter tuning, the model showcased a commendable ROC-AUC score of 0.9999818147225285. This high score signifies the model's near-flawless capacity to discern between players likely to be drafted and those who aren't.

- **Optimized Model Performance**: Post the hyperparameter tuning phase, there was a slight enhancement in the model's performance, registering a ROC-AUC score of 0.9999825194526576. The optimal parameters that contributed to this score were: bootstrap set to False, max_depth set to None, min_samples_leaf set to 1, min_samples_split set to 2, and n_estimators set to 200.

- **Updated Model Performance**: After further refinement and adjustments in the new notebook, like defining classes and functions, and feature selection, the model achieved a ROC-AUC score of 0.9998198685488762. While this is a slight dip from the previous score, it's still an exceptional score indicating a high level of accuracy.

- **Performance Analysis**: While the scores are exceptionally high, suggesting that the model has adeptly grasped the inherent patterns in the data, it's imperative to approach with caution. The score on Kaggle submission have also dropped. The reason for drop in scores is feature selection which uses the feature importance provided by Random Forest Model and inter-feature correlation, in comparison with the model trained without any feature selection. Ensuring that the model isn't overfitting to the training data is also crucial to maintain its generalization capabilities on unseen data.

3.b. Business Impact

From a business perspective, the model's predictions can have significant implications:

- **Draft Decisions**: NBA teams can leverage the model's predictions to inform their draft decisions, potentially identifying star players or hidden talents.

- **Player Development**: College players and their coaches can use the model's insights to focus on areas of improvement, increasing their chances of being drafted.

- **Potential Risks**: While the model's predictions are highly accurate, it's crucial to consider the potential risks of relying solely on the model. Incorrect predictions could lead to missed opportunities for teams or misplaced expectations for players.

3.c. Encountered Issues

Navigating through the experimentation phase often brings about unforeseen challenges. Here's a detailed account of the issues faced in the updated notebook:

- **Data Imbalance**: A significant challenge was the imbalanced distribution of the 'drafted' target variable. Such imbalances can skew the model's predictions, making it biased towards the majority class. In the updated notebook, the Synthetic Minority Over-sampling Technique (SMOTE) was employed to address this. By generating synthetic samples, SMOTE ensures a balanced representation of both classes, leading to a more unbiased model.

- **Feature Selection**: Determining the right set of features for the model was not straightforward. While certain features, like player statistics, naturally stood out due to their direct correlation with a player's likelihood of being drafted, others were not as evident. The updated approach leveraged the Random Forest's inherent feature importance mechanism. This not only streamlined the feature selection process but also ensured that the model was trained on the most relevant predictors.

- **Model Complexity and Overfitting**: The impressive ROC-AUC scores, though encouraging, also raised red flags about the model potentially memorizing the training data. Overfitting is a common pitfall in machine learning, where the model performs exceptionally well on training data but poorly on unseen data. To counteract this, the updated notebook incorporated various strategies. Regularization techniques were explored, and the model's hyperparameters were fine-tuned to strike a balance between accuracy and generalization. Additionally, the use of a validation set provided an unbiased evaluation of the model's performance, ensuring it was ready for real-world predictions.

By addressing these challenges head-on, the experimentation phase was not only about building a predictive model but also about understanding the intricacies of the data and the nuances of the modelling process.

4. FUTURE EXPERIMENT

4.a. Key Learning

While this week was focused on improving the Random Forest only with the help of feature selection and optimising the code by creating modules and publishing on Github with the help of cookiecutter, the experiment provided several valuable insights and learnings:

- **Optimised Coding:** The updated notebook showcases a more structured approach to coding. By encapsulating specific functionalities into classes and functions, the code becomes more maintainable, readable, and reusable. This modular approach also facilitates easier debugging and iterative development.
- **Model Performance:** The model achieved a high ROC-AUC score, indicating that player statistics from their college season are strong predictors for their likelihood of being drafted into the NBA.
- **Data Imbalance:** The target variable, 'drafted', had an imbalanced distribution. This could have led the model to be biased towards the majority class. The issue was addressed using the SMOTE technique, which was employed to balance the classes.
- **Feature Importance:** The model utilized a set of selected features based on their importance and correlation. Understanding the importance of each feature can guide future data collection and feature engineering efforts.
- **Overfitting Concerns:** The high ROC-AUC scores raised concerns about potential overfitting. Various regularisation techniques and model configurations were tested to ensure the model's generalisation capability.

4.b. Suggestions / Recommendations

Reflecting on the outcomes and insights from the experiment.

The following recommendations are proposed for future endeavours:

- **Ensemble Classification**: The power of ensemble methods is evident from the performance of the Random Forest model. However, there's potential to push the boundaries further. Techniques such as stacking, where predictions from multiple models are used as input to another model, can be explored. This layered approach can help in harnessing the strengths of different models, potentially leading to even better predictions.
- **Advanced Feature Engineering**: The updated notebook delves deeper into feature engineering, but the scores are dropping due to the feature selection, hence, there's always room for improvement. Considering exploring temporal features manually, capturing a player's performance trajectory over time. Such features can provide richer context and might be pivotal in predicting a player's draft status.

- **Model Diversification**: It's beneficial to explore a diverse set of models. Techniques like Gradient Boosting Machines (GBM) or even deep learning models like Neural Networks might offer unique insights and performance boosts. Each model comes with its strengths, and diversifying the model pool can help in capturing a wide array of patterns in the data.

- **Deployment Strategy**: The model's impressive performance warrants its deployment in real-world scenarios. A web-based tool or an API that, sports analysts, NBA teams, or even fans can use to predict draft outcomes can be envisioned. Such a tool can be a game-changer, offering data-driven insights to stakeholders.

- **Iterative Feedback Loop**: As the NBA draft unfolds and players' careers progress, this new data becomes invaluable. Establishing a feedback mechanism where the model is continuously updated with fresh data ensures that it remains relevant and accurate. This iterative approach, where the model learns from its past predictions, can lead to sustained accuracy and reliability.

By embracing these recommendations, future experiments can build upon the foundation laid by the current experiment, aiming for even higher levels of accuracy and business impact.