EXPERIMENT REPORT

Student Name	Monali Patil
Project Name	AT1 - Kaggle Competition
Date	01/09/2023
Deliverables	Notebook name: Patil_Monali-14370946-week3_EDA.ipynb Patil_Monali-14370946-week3_AdaBoost.ipynb Model name: AdaBoost Classifier Project Repo: https://github.com/MonaliPatil19/adv_mla_assignment1 Package Repo: https://github.com/MonaliPatil19/my_krml_package

---

1. EXPERIMENT BACKGROUND

Provide information about the problem/project such as the scope, the overall objective, expectations. Lay down the goal of this experiment and what are the insights, answers you want to gain or level of performance you are expecting to reach.

1.a. Business Objective

Explain clearly what is the goal of this project for the business. How will the results be used? What will be the impact of accurate or incorrect results?

Goal: The project involves building a binary classifier model that predicts whether a college basketball player will be drafted into the NBA league based on their statistics records. The dataset provided contains a wide range of features that illuminate players' performance during their college basketball season comprising 64 players' performance related attributes.

Application: This event captures the attention of sports commentators, fans, and enthusiasts who eagerly track the careers of college players and speculate about their chances of being selected by NBA teams. So, the model results will be used to assist sports commentators, fans, and teams in understanding the likelihood of a player's selection.

Impact: The model's accurate predictions can provide valuable insights for both players and teams, aiding decision-making during the NBA draft process. Additionally, this model will offer valuable insights to sports commentators, fans, and scouts, aiding them in predicting individual players' potential NBA draft prospects. Incorrect predictions, on the other hand, might lead to missed opportunities for players and teams, potentially impacting the overall performance and composition of the NBA teams.

1.b. Hypothesis	Present the hypothesis you want to test, the question you want to answer or the insight you are seeking. Explain the reasons why you think it is worthwhile considering it, Hypothesis: To test whether the statistical attributes of college basketball players from their college season can effectively predict whether they will be drafted into the NBA league. The key question would be, can a machine learning model accurately predict whether a college basketball player will be selected in the NBA draft based on their performance metrics? In this experiment, employing AdaBoost classifier, the aim is to assess the predictive power of these player statistics and evaluate whether they hold valuable insights into a player's likelihood of getting drafted. Rationale: Considering this hypothesis is worthwhile because the NBA draft is a significant event in a basketball player's career, and accurately predicting draft selection can have multiple benefits. For teams, it can provide a data-driven approach to identify potential talents and make informed decisions during the draft process. For players, it offers insights into the factors that influence their selection, potentially guiding their training and career decisions. Additionally, for fans and commentators, accurate predictions can enhance engagement and discussions around player performances and draft prospects. The use of statistical attributes also aligns with the growing trend of data-driven decision-making in sports, making this experiment relevant and valuable.
1.c. Experiment Objective	Detail what will be the expected outcome of the experiment. If possible, estimate the goal you are expecting. List the possible scenarios resulting from this experiment. The expected outcome of this experiment would be the development of a binary classifier model utilizing AdaBoost Classifier algorithm that can accurately predict whether a college basketball player will be drafted to the NBA league based on their college season statistics. The model would be trained on a dataset of college players information and their performance statistics, and it would use AdaBoost Classification algorithm to identify patterns and make accurate predictions. The goal of this experiment would be to identify potential player for an annual event aimed at identifying the players to be drafted in the NBA league utilising AdaBoost Classifier algorithm. The potential scenarios arising from this experiment include the following situations: • The model accurately predicts players who will be drafted. This provides teams with valuable insights for selecting promising talents, potentially improving their performance and team success. • The model inaccurately predicts players who will be drafted. Teams may miss out on potential talents, impacting their competitiveness. False positives rate can lead to misallocation of resources. • If the model performs moderately, there's an opportunity to fine-tune it. However, the model's marginal performance would be similar to random guessing. • The model is not accurate enough to be useful for predicting players position, and the experiment is unsuccessful. In this case, the NBA team may need to explore other methods for identifying potential players for their annual event.

2. EXPERIMENT DETAILS

Elaborate on the approach taken for this experiment. List the different steps/techniques used and explain the rationale for choosing them.

2.a. Data Preparation

Describe the steps taken for preparing the data (if any). Explain the rationale why you had to perform these steps. List also the steps you decided to not execute and the reasoning behind it. Highlight any step that may potentially be important for future experiments

The players' statistical data was explored, cleaned, and prepared through the subsequent activities for the binary classification algorithm to be employed.

• Data Understanding

1] Loading Data: Imported data from the given CSV file into the panda's data frames to use and create a binary classification model.

2] Exploring Data: Examined and studied players' information using different pandas functions to comprehend and uncover patterns, aiming to identify prospective players for NBA draft selection.

* Additionally, to ensure the quality of the data to be utilized by the model analysis, conducted below activities.

- Handing missing/null values.
- Eliminating identifiers.
- Duplicate records.
- Processing 'ht' feature to derive suitable information.
- Data distribution of various features.
- Selecting appropriate features based on the correlation coefficient.
- Accessing if imbalance targets classes.

df.head(): Checking initial records of the dataset.
df.shape(): Verifying the dimension of the dataset.
df.columns: Identifying attributes name.
df.info(): Assessing the summary information of the attributes of the dataset.
df.describe(include='all'): Examining statistical summary information for all variables of the dataset across different data types.
df.isnull().sum(): Examining whether there are any null values in the dataset.
df.duplicated().sum(): Identifying whether there are any duplicate values in the dataset.

• Data Preparation

3] Handling Missing Values

A significant number of 33 attributes in the training dataset and 24 in the testing dataset possess missing values, necessitating addressing to ensure data readiness for modelling as machine learning algorithms are incapable of processing features with missing values.

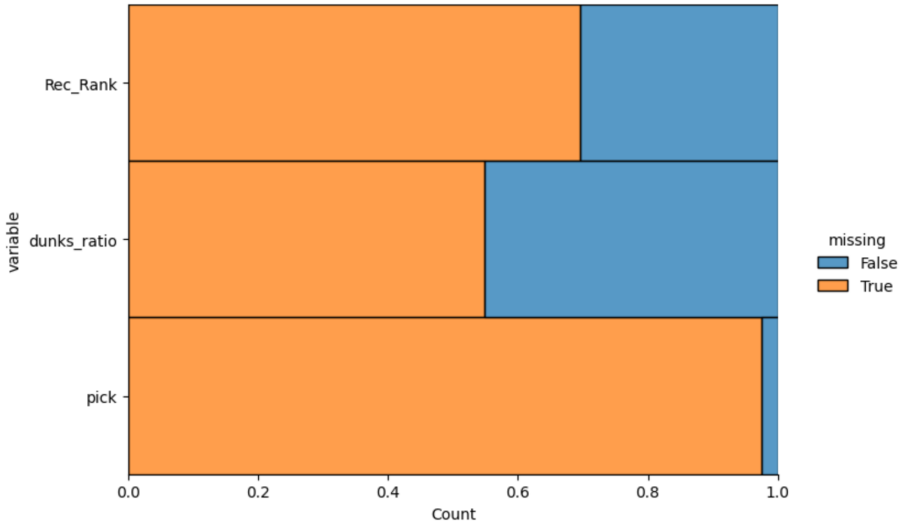

Figure 1: Attributes with majority of missing values.

* Among the features with missing data, the following three attributes exhibit notably high levels of missing values, exceeding 50%.

- Rec_Rank: Training -> 69.55% (39055/56091) and Testing -> 71.15% (3536/4970)
- dunks_ratio: Training -> 54.89% (30793/56091) and Testing -> 54.64% (2717/4970)
- pick: Training -> 97.64% (54705/56091) and Testing -> 98.98% (4921/4970)

* Therefore, it is practical to exclude these features in order to prevent potential biases in the model arising from imputation.

* Given that the remaining features have missing values comprising less than 2%, it is a reasonable approach to fill these missing values using the mean for numerical attributes and the mode for categorical attributes.

4] Processing 'ht' feature to derive suitable information

Please refer to 2.b Feature Engineering section.

5] Verifying and Removing Identifiers

* Attributes 'player_id' and 'num' are unique identifiers for each basketball player, and its inclusion in the analysis can lead to overfitting, where the model fits to these specific values rather than the underlying generalized patterns in the sportsman's records.

* Additionally, 'team' and 'conf' attributes representing the names of teams and conferences, are overly specific and do not contribute to the model's general learning.

* Therefore, removed 'player_id', 'num', 'team' and 'conf' features from the datasets.

6] Analysing correlation between predictors and target variable

* Performing correlation analysis to determine which features would be most valuable for building a predictive model for the NBA draft prediction task.

* Correlation using heatmap, calculates the correlation between each feature and the target variable 'drafted' to understand how much they influence the target. Features with high correlations to the target variable are potentially valuable and are selected as predictors.

7] Accessing if imbalance targets classes

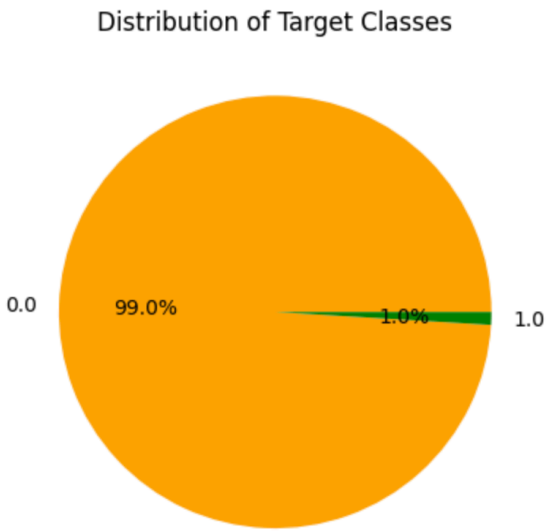

Figure 2: Target classes distribution.

* The pie chart above illustrates a significant class imbalance within the dataset. The majority of observations are attributed to a single target class, representing players who have not been drafted denoted by the value 0.

* Performed Oversampling with SMOTE (Synthetic Minority Over-sampling Technique) method to address class imbalance by generating synthetic observations for the minority class.

* It works by creating synthetic samples that are similar to existing observations in the minority class without duplicating the existing observations, rather it generates new observations by interpolating between existing ones.

8] Features Scaling

* Using feature scaling prevents the algorithm from prioritizing high-value features over other more informative ones. It ensures uniformity in feature values, enabling the algorithm to learn generalized patterns from all features for accurate player identification and predictions.

* Employed StandardScaler method because it maintains the features data distribution's shape and retains outliers by scaling data using the mean of 0 and standard deviation of 1 across the entire dataset, rather than for individual data points.

9] Hyperparameters Tuning

* Employed Hyperparameter Tuning to determine optimal values for the hyperparameters of the AdaBoost Classifier algorithm to further address slight overfitting nature of the mode.

The below are essential measures that could hold significance for any future classification experiments.

1. In order to avoid the model from overfitting on specific data points and to facilitate its learning of generalized patterns, it is crucial to remove any identification attributes.

	2. Thoroughly assessing and managing missing values is essential to prevent introducing biases into the model. 3. Choosing the appropriate feature scaling method among MinMaxScaler, MaxAbsScaler, or StandardScaler should depend on the presence of natural data ordering and considerations like outlier preservation. 4. In classification tasks, particularly when dealing with imbalanced data, it's crucial to maintain the same target variable's class frequencies in each set accurately.
2.b. Feature Engineering	Describe the steps taken for generating features (if any). Explain the rationale why you had to perform these steps. List also the feature you decided to remove and the reasoning behind it. Highlight any feature that may potentially be important for future experiments Processing 'ht' feature to derive suitable information. * The 'ht' attribute, which represents the height of players, contains alphanumeric values in a date format which is significant in the basketball game and for selection of the players. When examining the unique value patterns, it becomes evident that values like 'Jun,' 'Jul,' and 'Aug' cannot represent height in months. Instead, it's reasonable to infer that 'Jun' corresponds to 6 feet, 'Jul' to 7 feet, and so forth. * As a result, transformed the date values into their corresponding numerical representations and further converted them into a new numerical attribute named 'ht_cm,' representing height in centimeters, to facilitate easier analysis and processing. * In order to build machine learning models that necessitate numerical inputs the categorical feature 'ft_inch' is transformed into numerical values and computed height in 'ft_cm' which represents players height in centimeters.
2.c. Modelling	Describe the model(s) trained for this experiment and why you choose them. List the hyperparameter tuned and the values tested and also the rationale why you choose them. List also the models you decided to not train and the reasoning behind it. Highlight any model or hyperparameter that may potentially be important for future experiments The target variable is binary, containing 0 signifying players who are not selected, and 1 representing drafted players. With our goal to predict outcomes within these two classes, a Binary Classification model is appropriate, aligning with the objective of categorizing players into these distinct groups. Employed the AdaBoost Classifier algorithm in this experiment, which is popular and effective algorithm for binary classification task, improving the performance of weak learners, making it a valuable tool to precisely assess if the player will be chosen for the NBA league.

Hyperparameters Selected.

Hyperparameters Name	Values Tested
n_estimators:	1000, 50, 209
learning_rate:	1.0
algorithm:	SAMME
estimator:	DecisionTreeClassifier

* The above provided hyperparameter values were employed to address the minor overfitting observed in the model.

* Setting n_estimator hyperparameter specifies the upper limit on the number of iterations or weak learners to be trained.

* Utilising learning_rate hyperparameter to regulate the impact of each individual weak learner on the ultimate prediction.

* The algorithm hyperparameter allows the model to use the SAMME discrete boosting algorithm.

* The estimator hyperparameter designates the category of the weak learner to be employed, including options like decision stumps or decision trees.

Collectively, these tuned hyperparameter values optimize the AdaBoost classification model's performance to address the issue of slight overfitting observed in the model.

Because of time limitations, there wasn't enough oppourtinity to utilize an extra classification model.

3. EXPERIMENT RESULTS

Analyse in detail the results achieved from this experiment from a technical and business perspective. Not only report performance metrics results but also any interpretation on model features, incorrect results, risks identified.

3.a. Technical Performance

Score of the relevant performance metric(s). Provide analysis on the main underperforming cases/observations and potential root causes.

Baseline AUROC Performance: 0.5

* In the 1st experiment, the AUROC performance score of 0.9883 on the validation set is slightly below the score of 0.9905 achieved on the training set, suggesting that the model is exhibiting a very minor degree of overfitting.

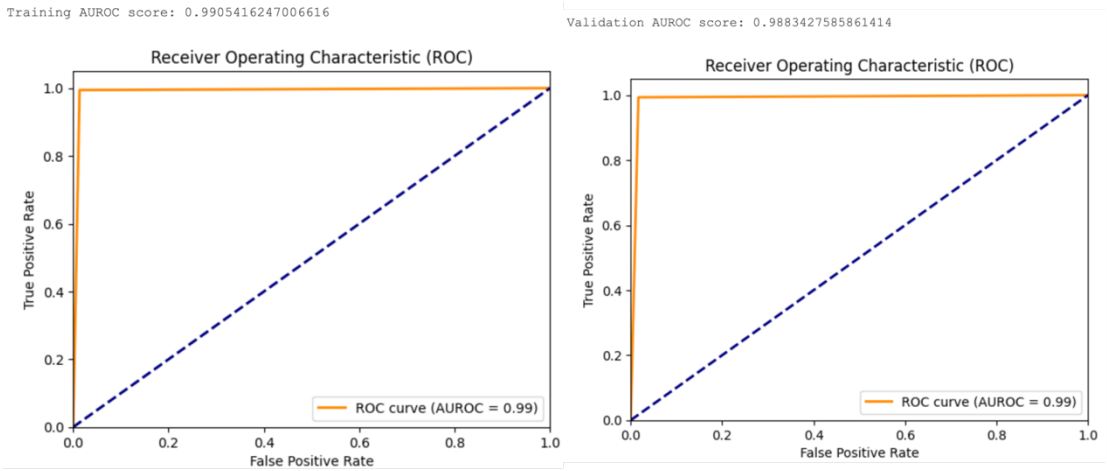

Figure 3: ROC plot from the 1st experiment.

* Furthermore, examining the ROC curves for both the training and validation datasets, which visually illustrate the balance between True Positive Rate (TPR) and False Positive Rate (FPR) at various threshold values, it indicates that the model shows a relatively consistent performance on both the datasets.

* Therefore, in the 2nd experiment, employed Hyperparameter for the AdaBoost Classifier models, aiming to mitigate the minor overfitting observed in the model.
Below are the Hyperparameters values of the AdaBoost classifier:

Experiment 2: estimator=DecisionTreeClassifier(), n_estimators=50, learning_rate=1.0, algorithm='SAMME'

Experiment 3: estimator=DecisionTreeClassifier(max_depth=487, min_samples_leaf=14, class_weight='balanced', criterion='entropy', max_features = 'log2'), n_estimators=209, learning_rate=1.0, algorithm='SAMME'

* Experiment 2: Training - 1.0, Validation 0.9844
* Experiment 3: Training - 1.0, Validation 0.9945

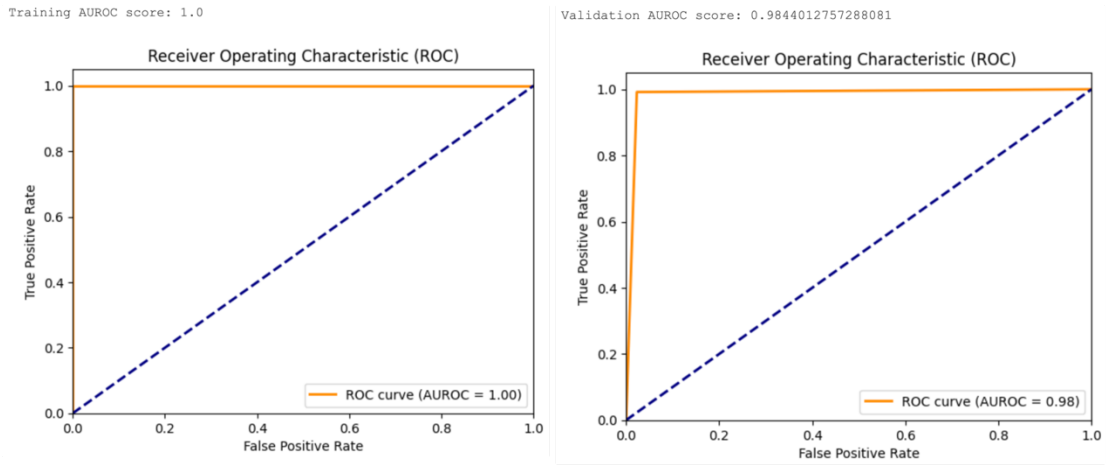

Figure 4: ROC plot from the 2nd experiment.

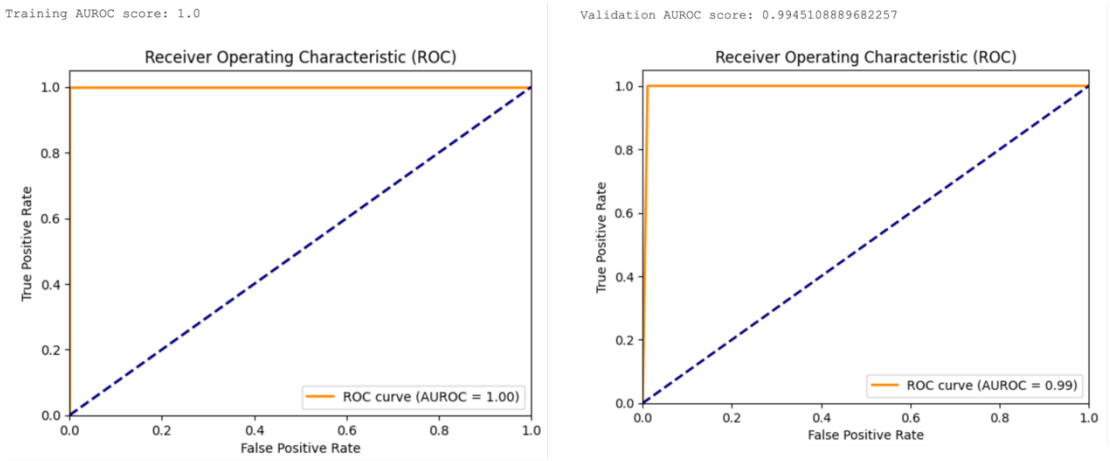

Figure 5: ROC plot from the 3rd experiment.

* There is slight difference in AUROC scores between the training and validation datasets illustrating that the models from both the experiments, with its hyperparameters, shows consistent performance across datasets, with marginal distinctions.

* The ROC curves for both the training and validation datasets in both experiments are relatively closer to the upper-left corner, indicating that the model exhibits subtle variations in its behavior between the two datasets. Therefore, the models exhibits a mild form of overfitting.

* However, this slight difference is marginally greater than the AUROC performance score of experiment 1, indicating that there is a possibility of prospective players being misclassified and overlooked during the NBA league selection process.

3.b. Business Impact

Interpret the results of the experiments related to the business objective set earlier. Estimate the impacts of the incorrect results for the business (some results may have more impact compared to others)

The features that have a significant impact on the selection of NBA league players include 'dporpag,' 'porpag,' 'bpm,' 'gbpm,' 'adjoe,' 'ogbpm,' 'stops,' 'twoPA,' 'obpm,' 'twoPM,' 'dunksmade,' 'dunksmiss_dunksmade,' 'pts,' 'dreb,' 'FTM,' 'FTA,' 'treb,' 'blk,' 'Min_per,' 'usg,' 'ht_cm,' 'Ortg,' and 'midmade_midmiss.'

Utilizing the insights from important features, the Adaboost classifier in the 1st experiment achieved the better performance than the models from 2nd and 3rd experiment with hyperparameters, which indicates a slight tendency toward overfitting, and the model may not be sufficiently generalized to make accurate predictions for potential draft selections on new, unseen data.

3.c. Encountered Issues	List all the issues you faced during the experiments (solved and unsolved). Present solutions or workarounds for overcoming them. Highlight also the issues that may have to be dealt with in future experiments. During the execution of pytest within the package, an issue was encountered that was resolved with the help of a query posted in the discussion forum.

---

4. FUTURE EXPERIMENT	
Reflect on the experiment and highlight the key information/insights you gained from it that are valuable for the overall project objectives from a technical and business perspective.	
4.a. Key Learning	Reflect on the outcome of the experiment and list the new insights you gained from it. Provide rationale for pursuing more experimentation with the current approach or call out if you think it is a dead end. The features that have a significant impact on the selection of NBA league players include 'dporpag,' 'porpag,' 'bpm,' 'gbpm,' 'adjoe,' 'ogbpm,' 'stops,' 'twoPA,' 'obpm,' 'twoPM,' 'dunksmade,' 'dunksmiss_dunksmade,' 'pts,' 'dreb,' 'FTM,' 'FTA,' 'treb,' 'blk,' 'Min_per,' 'usg,' 'ht_cm,' 'Ortg,' and 'midmade_midmiss.' Utilizing the insights from important features, the Adaboost classifier in the 1st experiment achieved the better performance than the models from 2nd and 3rd experiment with hyperparameters, which indicates a slight tendency toward overfitting, and the model may not be sufficiently generalized to make accurate predictions for potential draft selections on new, unseen data.
4.b. Suggestions / Recommendations	Given the results achieved and the overall objective of the project, list the potential next steps and experiments. For each of them assess the expected uplift or gains and rank them accordingly. If the experiment achieved the required outcome for the business, recommend the steps to deploy this solution into production. Before considering the deployment of the model in the operational environment, it's advisable to evaluate the performance of alternative algorithms using the Feature Importance information. Assessing all the trained models from the three week experiments, it is clear that the AdaBoost classifier from the 1st experiment of this third week, utilizing Feature Importance information, stands out as the most efficient model and is suitable to deploy for predicting the selection of potential players in the NBA league.