EXPERIMENT REPORT

Student Name	Shi Wu
Project Name	Part2- LightGBM+feature_engineering
Date	23/08/2023
Deliverables	<notebook name> < LightGBM+feature_engineering> <other>

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? This notebook will use LightGBM to predict whether a college basketball player will be drafted into the NBA based solely on season statistics. This model helps NBA teams decide who to watch in person, plan for the draft, and see if their own ratings match the predictions. A good model helps teams scout smarter, but a bad one might lead them to miss out on talent or waste resources.
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: College basketball players with higher season statistics are more likely to be drafted into the NBA. Question: Do college basketball players' season statistics directly influence their chances of being drafted into the NBA? We aim to determine the correlation between a college player's season performance and their probability of being drafted into the NBA. Understanding the link between college statistics and draft probability can greatly optimize the processes and decisions related to scouting, investing, and planning in the NBA.

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 primary expected outcome is to determine a quantifiable relationship between college basketball players' season statistics and their likelihood of being drafted into the NBA. It's difficult to identify a specific objective given the wide range and abundance of factors that affect the draught. But let's imagine that based on a player's season stats, we want to be able to forecast with 80% accuracy whether they would be selected. Possible scenarios resulting from this experiment include: 1. High Correlation: The model might find that players' season statistics have a strong correlation with their draft status. In this scenario, NBA teams can significantly rely on statistical data to make their draft decisions. 2. Moderate Correlation: The model could indicate that while statistics play a role in draft decisions, other factors (like a player's physical attributes, potential, or team needs) also carry significant weight. In this case, teams would use the model's predictions as supplementary data in their overall scouting and decision-making processes. 3. Low or No Correlation: The model may determine that there's little to no correlation between season statistics and draft status. This would mean that while statistics are informative, many other aspects influence draft decisions, and teams should not heavily rely on stats alone. 4. Overfitting or Inconsistencies: The model might show a high accuracy during training but perform poorly on new, unseen data. This suggests that while it might have learned patterns from the historical data, it may not be generalized enough for real-world predictions. 5. Variable Importance: The model might reveal that only certain statistics (e.g., points per game or efficiency ratings) are predominantly influencing draft

---

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 1. The dataset is loaded and explored to understand the data structure, missing values, and basic statistics. 2. We dropped those variables with more than 50% missing values, replaced all the missing value with 'unknown'. 3. We dropped one variable with single values
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 1. OrdinalEncoder coding was applied to all discrete variables; 2. We add the following features: avg_points_per_game(pts/GP), Ortg_diff_from_mean(ortg-ortg_mean), usg_Ortg_interaction(usg*Ortg), Ortg_square(ortg^2), pts_rolling_avg 3. We dropped 12 features with correlation greater than 0.95 4. We keep the features that cumulative importance of 0.99

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 LightGBM is a gradient boosting framework known for its fast training speed and efficiency, especially with large datasets. It can handle categorical features directly, supports parallel learning, and offers built-in regularization to prevent overfitting. Given these advantages, it's a popular choice for many machine learning tasks, especially when time and computational resources are crucial. We used grid search with 5 fold cross-validation to find the best model. We tuned the following hyperparameters for LightGBM model: 1. classifier__learning_rate: It represents the learning rate or step size used during the gradient boosting process. Smaller values result in slower convergence but potentially better performance, while larger values can lead to faster convergence but possibly worse performance. The grid includes three values: 0.01, 0.1, and 1. 2. classifier__n_estimators: This parameter specifies the number of boosting rounds or trees in the ensemble. More trees can result in better performance but also increase the risk of overfitting. The grid includes five values: 20, 50, 100, 200, and 500. 3. classifier__num_leaves: It defines the maximum number of leaves in each tree. A higher number of leaves can model more complex relationships but may also lead to overfitting. The grid includes three values: 31, 62, and 93. 4. classifier__reg_alpha: This parameter adds L1 regularization to the model, which can help prevent overfitting by encouraging sparsity in the learned weights. The grid includes four values: 0.0, 0.1, 0.5, and 1.0. 5. classifier__reg_lambda: similar to classifier__reg_alpha, this parameter adds L2 regularization to the model. L2 regularization discourages large weights in the model, making it less likely to overfit. The grid includes four values: 0.0, 0.1, 0.5, and 1.0. In the second experiment we explored some feature engineering and feature selection, we will try to implement deep learning to achieve higher performance in further experiments.

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. The result AUC is 0.98785, which is an slight improvement of our previous score of 0.98577. Some derived variables may have had an effect during the classifier training 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 results strongly suggest that type(1942), rimmde_rimmiss(1260), FT_per(1134), GP(1060) are significant factors influencing NBA draft decisions, The top features in this dataset focus on game efficiency (like shooting close to the basket and overall offensive efficiency), experience (games played and possibly years of experience), and game situations or categories (type). These features provide a comprehensive understanding of a player's or team's performance and their ability to succeed in various game situations. Incorrect results may lead to the evaluation of players based on a wrong system, which may lead to the team assigning an unreasonable development plan and wasting resources eventually.
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 deal with in future experiments. 1. Failed to combine featureselector into pipeline. So it was hard to apply uniform feature engineering (like onehot encoder) on both train and test dataset. 2. We added derivative features manually both on train and test dataset ensure the pipeline runs normally on test dataset.

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. We used feature importance to interpret the result of model, and the most three important features are: 1. type (1942): The "type" feature's high importance score indicates that the phase or segment of the basketball season has a significant impact on the analysis or prediction being conducted. Different segments of the season come with different pressures, competitive levels, and implications, which can influence player and team performances. Differentiating between these segments can provide nuanced insights into how a player or team performs under various conditions. 2. rimmade_rimmiss (1260): This probably represents the ratio or difference between made shots near the basket (like layups or dunks) and those that were missed. The efficiency of a player or team in making shots close to the basket is crucial, as these shots usually have a higher success rate. This feature's prominence indicates the importance of capitalizing on close-range opportunities in basketball. 3. FT_per (1134): Free throws offer a chance to score without any defensive pressure, so a player's or team's ability to consistently make free throws can significantly affect the game's outcome. A high free throw percentage can be indicative of good shooting fundamentals and can be a crucial factor, especially in close games. 4. GP (1060): The number of games a player or team has participated in can be a direct indicator of experience and consistency. Players or teams that have played more games might have more data points, which can provide a more comprehensive understanding of their performance trends. Ensemble model is complex and hard to understand (like some ensemble methods or deep learning models), exploring model interpretability tools can be beneficial. This is especially important if the model is to be presented to stakeholders or used in decision-making processes. In the subsequent experiments we will apply more
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. In this experiment we used LightGBM with feature engineering obtained a improved model with AUC 0.98785, as well as the importance of features. Since there exist bottleneck in current model, thus we may try train deep learning model regardless of insufficient samples.