# EXPERIMENT REPORT

| | |
|---|---|
| **Student Name** | Saumya Bhutani |
| **Project Name** | AT 1 Part C |
| **Date** | 01/09/2023 |
| **Deliverables** | <SaumyaBhutani_AT1_PartA_14360820><br><XGBoost Automated Hyperparameter Tuning and Regularization><br><GitHub Link: https://github.com/bhutanisaumya/Adv_MLA.git> |

---

| 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** | The project aims to create a predictive model to forecast whether a college basketball player will get drafted into the NBA based on their performance matrices.<br>Using a machine learning model to predict which players are more likely to be drafted, the business can optimise their efforts for NBA teams by helping them allocate resources effectively for scouting, assisting analysts in player evaluation, aiding player agents in advising their clients, engaging fans and media with data-driven discussions, and guides strategic draft decisions. The project's success is measured by how well the model can predict draft outcomes, evaluated through the AUROC metric.<br><br>In this experiment, I aim to determine how well a boosting model with Automated Hyperparameter Tuning and regularization can predict the drafting probabilities using multiple independent features.<br><br>Impact of accurate result:<br>If the model accurately identifies drafting of players then the business can optimize the accurate results from the predictive model positively to influence draft strategies, player decisions, scouting efficiency, and fan engagement.<br>Accurate results will increase the model's reliability, and the model is more likely to be used to make better predictions.<br><br>Impact of incorrect result:<br>If the model incorrectly identifies players less likely to be drafted, then the incorrect results could lead to suboptimal decisions, misallocation of resources, and potential disappointment among stakeholders.<br>The model will not be reliable and cannot be used to make predictions. Businesses' goal of predicting the players to be drafted in NBA will remain unfulfilled as the model predicts the incorrect result. |

| | |
|---|---|
| **1.b. Hypothesis** | Hypothesis: The XGBoost model can effectively predict the players to drafted in NBA( target) by analysing various independent features such as player performance metrics, technicalities of player's game etc.<br><br>The hypothesis I want to test is whether specific performance statistics of college basketball players can be used to predict their likelihood of them being drafted into the NBA. Specifically, I want to explore if metrics such as scoring efficiency, rebounding, assists, defensive metrics, and overall playing time, along with other metrics, significantly impact a player's draft status.<br><br>Considering this hypothesis is worthwhile because it can potentially bring data-driven insights to the NBA draft process, benefiting teams, players, scouts, and fans. It aligns with the industry's increasing reliance on analytics and addresses the complexities of selecting and predicting the success of future NBA players. The insights gained from this project can contribute to more informed decisions and a deeper understanding of the factors influencing the transition to professional basketball. |
| **1.c. Experiment Objective** | I expect the model to give results in favour of my Hypothesis.<br><br>However, it may have the following possible outcomes:<br><br>1. Positive outcome: The XGBoost classifier can identify significant predictors. The model's roc au score is high. The model can be used by the business to predict potential NBA players.<br><br>2. Negative outcome: The XGBoost classifier cannot identify significant predictors. The model's roc au score is low. The business cannot use the model to predict potential NBA players.<br><br>3. Inconclusive outcome: The XGBoost classifier identifies some significant predictors, but the model's roc au score is low or insignificant enough to be actionable. |

---

| **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**

For this experiment, as I am using XGBoost model,

I performed the following data preparation steps:

1. Checked for null values across all the features. It was identified that :

    a. The features ‘pick’ and 'Rec_Rank' have a high percentage of missing values of more than 97% and 69%, respectively. Therefore, I did not consider these features for my analysis.

    b. The columns ‘ast_tov’, ‘rimmrade’, ‘rimmade_rimmiss’, ‘mid_made’, ‘midmade_midmiss’, ‘dunksmade’, ‘dunksmiss_dunksmade’, 'drtg', 'adrtg', 'dporpag', 'stops', 'bpm', 'obpm', 'dbpm', 'gbpm', 'mp', 'ogbpm', 'dgbpm', 'oreb', 'dreb', 'treb', 'ast', 'stl', 'blk' and 'pts' having missing values were substituted with their mean values.

    c. For the columns having ratios, the missing values were calculated using the below formulas.
        - rim_ratio = rimmade / rimmade_rimmiss
        - mid_ratio = midmade / midmade_midmiss
        - dunks_ratio = dunksmade / dunksmiss_dunksmade

    d. The columns 'yr' and 'num' contain a lot of irregularities. Since the data is unreliable. Thus I have decided to remove these columns.

2. Data splitting
   I separated the dependent feature (‘drafted’) into y. I have split the training set in 80:20 into the training and validation set. I have specifically split this ratio so there is enough training data. I will be using the validation set to calculate roc_auc_score.

   The X_feaures contains player performance metrics, technicalities of the player and draft attributes. I have considered them on the basis of the correlation matrix for the analysis.

   X_features: 'porpag', 'dunksmade', 'dunksmiss_dunksmade', 'dporpag', 'twoPM', 'FTM', 'FTA', 'twoPA', 'midmade', 'pts', 'midmade_midmiss', 'rimmade', 'stops', 'dreb', 'treb', 'rimmade_rimmiss', 'blk', 'mp', 'Min_per', 'stl', 'oreb', 'bpm', 'ast', 'gbpm', 'obpm', 'TPM', 'ogbpm', 'adjoe', 'TPA', 'usg', 'GP', 'dgbpm', 'dunks_ratio', 'dbpm', 'Ortg', 'FT_per', 'TS_per', 'rim_ratio', 'AST_per', 'eFG', 'twoP_per', 'TP_per', 'mid_ratio', 'ast_tov', 'DRB_per', 'blk_per', 'stl_per', 'ORB_per', 'ftr', 'ht', 'year', 'pfr', 'TO_per', 'drtg', 'adrtg'

3. Resampling of Training data
   The dependent feature is highly imbalanced, wherein 0 is more than 99%, and 1 is less than 1%. Class 1 represents if the player is drafted in NBA, and class 0 represents the player is not drafted. Data resampling is important here as there is a significant class imbalance, as one class represents only a small fraction of the dataset. Thus, the model can be biased towards the majority class, making it difficult to predict the minority class accurately.

| | |
|---|---|
| | Resampling the Minority class,<br>I have performed oversampling of minority class(1) using SMOTE(Synthetic Minority Over-sampling Technique) in the training dataset only. |
| **2.b. Feature Engineering** | Dropped features: The features ‘pick’ and 'Rec_Rank' have a high percentage of missing values of more than 97% and 69%, respectively. Therefore, I did not consider these features for my analysis.<br>The columns 'yr' and 'num' contain a lot of irregularities. Since the data is unreliable. Thus I have decided to remove these columns<br><br>Categorical features: I have not considered categorical features such as ‘team’, ‘conf’.<br><br>Handled ‘ht’ column: The datatype was object and the values were represented in date format. eg. 2-June-2023. I converted them in int datatype (feet.inches)as described below:<br>2/06/2023 —> 2062023 —> 6022023 (M/d/Y) —-> 6.02 (removed the year i.e., M/d)<br>Which implies to 6 Feet 2 Inches<br><br>Feature scaling: I have performed feature scaling on all independent features (training, validation and testing sets) as It ensures that all features contribute to the model fairly and that the algorithms operate effectively regardless of the original scale of the data. It improves model performance, convergence, and the ability to make meaningful predictions by removing any distortions caused by feature scales |
| **2.c. Modelling** | XGBoost Model Description<br>For this experiment, an XGBoost (Extreme Gradient Boosting) model was trained to predict whether a college basketball player will be drafted to join the NBA league based on their statistics. XGBoost is a popular ensemble learning algorithm known for its high performance and effectiveness in handling complex relationships in the data.<br><br>I opted for XGBoost because of its robust performance and adaptability. XGBoost offers built-in support for both L1 and L2 regularization, enhancing the model's ability to generalize, a feature that AdaBoost lacks. I specifically employed L2 regularization to boost the model's generalization capabilities. Although I experimented with LightGBM as an alternative, it did not yield satisfactory performance.<br><br>Automated Hyperparameterization using the Grid Search<br>colsample_bytree=0.8, learning_rate=0.01, max_depth=4, min_child_weight=1, n_estimators=400, subsample=0.8, reg_lambda = 0.5 |

| 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** | Automated Hyperparameterization generated using:<br>Grid Search<br><br>colsample_bytree=0.8, learning_rate=0.01, max_depth=4, min_child_weight=1, n_estimators=400, subsample=0.8, reg_lambda = 0.5<br><br>ROC AUC score on validation set: 0.9437<br><br>A ROC AUC score of 0.9437 on the validation set using XGBoost with hyperparameter tuning indicates that the model is performing well in terms of its ability to discriminate between positive and negative classes. The ROC AUC score measures the area under the receiver operating characteristic curve, which quantifies the model's ability to distinguish between true positive and false positive rates across different probability thresholds.<br><br>In practical terms, based on their features and statistics, the XGBoost model effectively separates the drafted players from the non-drafted players.<br><br>The predictions on the probability of players being drafted or not have been made for the X_test_scaled for each player_id. |
| **3.b. Business Impact** | The model achieved a robust ROC AUC score of 0.9437, highlighting its ability to differentiate drafted and non-drafted college basketball players based on performance metrics. However, it's essential to address both false positives (missed opportunities) and false negatives (overlooking talent) for credibility. The model's outcomes impact team choices, player prospects, fan expectations, and team performance, emphasizing the need for ongoing model refinement and assessment. |
| **3.c. Encountered Issues** | Throughout the experiment, several challenges were encountered and addressed, while some remain open for future consideration:<br><br>Key Steps:<br><br>1 Used SMOTE. Applied Min-Max scaling to standardize feature values, promoting faster model convergence and better performance while preventing any single feature from dominating the learning process.<br><br>2. Developed a strategic approach to address missing data. Median imputation was employed for ratio features, minimizing the influence of outliers on imputed values and ensuring data integrity.<br><br>3. Streamlined hyperparameter optimization using GridSearchCV, systematically exploring hyperparameter combinations to identify the optimal model configuration. Integration of L2 regularization was employed to combat overfitting and enhance model |

| | |
|---|---|
| | generalization.<br><br>4. Conducted a thorough evaluation of multiple algorithms, such as XGBoost, AdaBoost, and LightGBM.<br><br>Collectively, these data preprocessing and model selection strategies laid the foundation for an accurate and robust predictive model, combining feature scaling, effective missing data handling, systematic hyperparameter optimization, and informed algorithm choice to create a powerful predictive analytics solution. However, there is still room for improvement. |

---

| **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** | The experiment yielded valuable insights into NBA draft outcome prediction through machine learning. Notably, XGBoost exhibited robust predictive capabilities with high ROC AUC scores. Significant correlations between key player statistics and draft outcomes were identified. Hyperparameter tuning and missing data handling further improved model performance and data quality. The approach holds promise and invites further exploration. Future avenues include investigating ensemble methods, addressing class imbalances, and advanced feature engineering to potentially enhance results. Emphasizing model interpretability and real-world deployment readiness are crucial for future research. This approach serves as a stepping stone, and ongoing refinement and adaptability are vital for practical applications. |
| **4.b. Suggestions / Recommendations** | Steps to be performed in future:<br><br>Ensemble Models: Experiment with ensemble methods like stacking or blending different models to further enhance predictive accuracy. Expected Uplift: Moderate.<br>Feature Engineering: Explore more advanced feature engineering techniques, leveraging domain knowledge to create novel features that capture player performance nuances. Expected Uplift: Moderate to High.<br>Model Interpretability: Implement model interpretability techniques like SHAP (SHapley Additive exPlanations) to gain insights into feature importance and decision-making. Expected Uplift: Moderate.<br>Validation on Unseen Data: Test the model's performance on unseen data from subsequent NBA draft years to assess generalisation. Expected Uplift: Moderate.<br><br>Refining the feature engineering, which would enhance the predictions of the model. Boosting models with careful tunning are likely improve the model's performance further.<br><br>Deployment Steps: |

| | |
|---|---|
| | Model Evaluation: Ensure the model meets validation and unseen data performance thresholds.<br>Interpretability: Implement techniques to understand feature importance and decision logic.<br>Infrastructure Setup: Prepare the deployment environment with the necessary packages.<br>Scalability Test: Assess the model's scalability for real-time or batch predictions.<br>Model Deployment: Choose a deployment approach (API, microservices, cloud).<br>Monitoring & Maintenance: Continuously monitor, retrain, and update data.<br>Feedback Loop: Gather insights and iterate on model improvements with stakeholders.<br>Deploy if the model achieves the required outcomes and shows robustness in testing.<br>Maintain regular monitoring and updates to align with dynamic NBA draft trends. |

GitHub Link: https://github.com/bhutanisaumya/Adv_MLA.git