CSE 158 Assignment 2 Report
Exploring Predictive Models on PUBG Dataset

Chengyuan Mao

University of California, San Diego
c1mao@ucsd.edu

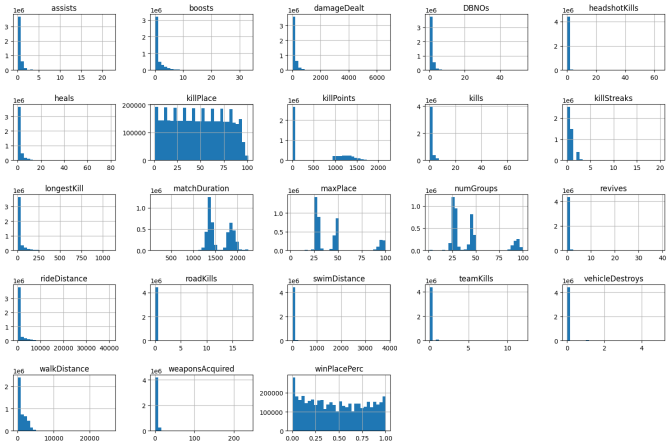
***Abstract*—This report presents an exploratory analysis and predictive modeling task on the PUBG dataset. I analyze the dataset's properties, propose a predictive model to estimate players' win placement percentage based on in-game statistics, compare it with baselines, and discuss my findings and conclusions. My approach demonstrates that key features like kills, damage dealt, and distance traveled significantly impact player performance, paving the way for better understanding of success factors in battle royale games.**

I. Dataset Description and Exploratory Analysis

I have selected the PlayerUnknown's Battlegrounds (PUBG) dataset for my study. This dataset contains game statistics from over 4.4 million observations, capturing various in-game activities and performance metrics such as kills, assists, damage dealt, distances traveled, and the final placement percentage (winPlacePerc).

In my exploratory data analysis (EDA), I performed basic statistical analyses and visualizations to understand the dataset's properties.

- **Basic statistics:** The dataset consists of a large number of features with varying scales and distributions. Key observations include high variance in features like damageDealt, walkDistance, and kills, indicating a wide range of player performance.
- **Missing values:** I checked for missing values in the dataset and found that there is only one missing value in the winPlacePerc column. I decided to drop this observation as it would not significantly impact the analysis.
- **Data distributions:** I visualized the distributions of features using histograms and it revealed that most features are skewed, suggesting the presence of outliers or non-normal distributions.



- **Correlation Analyses:** I found significant positive correlations between walkDistance, kills, damageDealt, and winPlacePerc, indicating that better in-game performance is associated with higher final placement.
- **Feature Selection:** Based on the correlation analysis, I decided to drop the features with low correlations with the target variable (winPlacePerc), such as roadKills, teamKills, killPoints, rankPoints, and winPoints to reduce noise and improve model performance.

Interesting Findings:

- Players who cover more distance tend to achieve better final placements.
- Certain features had negligible impact on the outcome and were excluded from further analysis.

II. Predictive Task and Evaluation

My predict task is to predict the player's final placement percentage (winPlacePerc) based on their in-game statistics. This is a regression problem where I aim to model the relationship between player actions and their success in the game.

A. Evaluation Strategy

Performance Metric: I will use Mean Squared Error (MSE) to evaluate model performance, providing a measure of the average squared difference between predicted and actual winPlacePerc values.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{1}$$

Where:

- n = number of samples/observations
- y_i = true/actual value for the i-th sample
- $\hat{y}_i$ = predicted/estimated value for the i-th sample

Train-Test Split: The dataset is split into training (80%) and validation (20%) sets to assess the model's generalizability.

Baselines for Comparison:

- Linear Regression

$$\min_{\beta}\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 \tag{2}$$

  Where:
  - y_i is the target variable
  - x_{ij} are the features
  - β_0 is the intercept
  - β_j are the regression coefficients

- Ridge Regression

$$\min_{\beta}\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 + \lambda\sum_{j=1}^{p}\beta_j^2 \tag{3}$$

  Where:
  - λ is the regularization parameter (controls the strength of the penalty)
  - The additional term $\lambda\sum_{j=1}^{p}\beta_j^2$ shrinks the coefficients towards zero

- Lasso Regression

$$\min_{\beta}\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 + \lambda\sum_{j=1}^{p}|\beta_j| \tag{4}$$

  Where:
  - λ is the regularization parameter
  - The additional term $\lambda\sum_{j=1}^{p}|\beta_j|$ can drive some coefficients exactly to zero

Validity Assessment: I compare the MSE scores of different models and analyze feature importances to understand which factors contribute most to the predictions.

III. Model Description

I experimented with various models to identify the best predictor for winPlacePerc.

Linear Models:

- **Linear Regression:** Provided a baseline but was limited in capturing complex relationships between features and target variable.
- **Ridge Regression:** Added L2 regularization to the linear regression to handle multicollinearity and reduce overfitting.
- **Lasso Regression:** Utilized L1 regularization for feature selection, reducing model complexity and improving interpretability.

Ensemble Models:

- **Random Forest:** Utilized an ensemble of decision trees to capture non-linear relationships and interactions between features.
- **LightGBM:** A gradient boosting framework that optimizes training speed and performance, handling large datasets efficiently.

Optimization and Challenges:

- **Hyperparameter Tuning:** Adjusted parameters like the number of estimators and regularization strength to optimize model performance.
- **Overfitting:** Observed in complex models; addressed through cross-validation and early stopping.
- **Feature Engineering:** Created new features (e.g., totalDistance, performanceScore) to enhance model capacity to capture player performance nuances.

Strengths and Weaknesses:

- **Linear Models:** Simple and interpretable but limited in capturing non-linear relationships.
- **Ensemble Models:** More complex and powerful, but prone to overfitting and require careful tuning.
- **LightGBM:** Efficient and scalable, providing high predictive accuracy with minimal computational resources.

IV. Related Literature

The PUBG dataset was provided by Kaggle and has been used in various studies. However, most existing works focus on player behavior analysis and outcome prediction using machine learning techniques. Research has demonstrated that in-game statistics can effectively predict player success.

My approach builds upon these studies by leveraging advanced algorithms and feature engineering to improve prediction accuracy. I extend the Literature by:

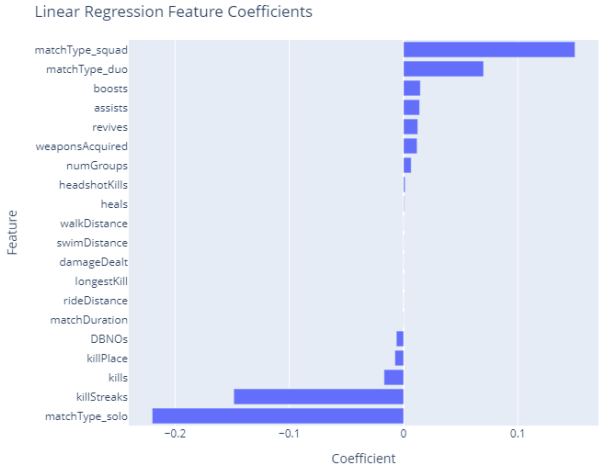
- Incorporating advanced models like LightGBM for improved prediction accuracy.
- Emphasizing feature engineering to derive meaningful insights from raw data.
- Providing a comparative analysis of different regression techniques on the PUBG dataset.

V. Results and Conclusions

A. Model Performance with Original Features

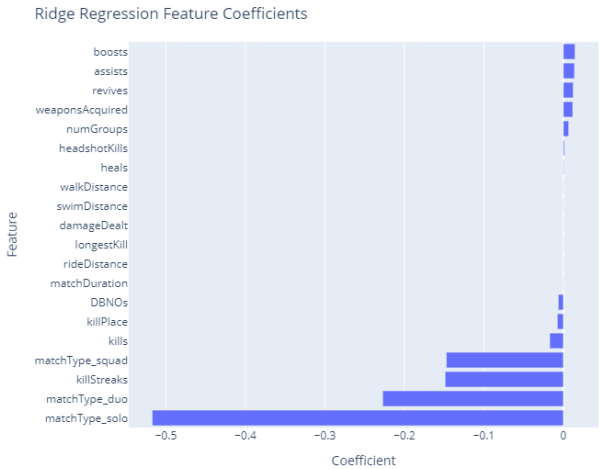
Linear Regression Model:

- Achieved an MSE of 0.014833 on the validation set, indicating a reasonable fit to the data.
- Identified key features like matchType_squad, matchType_duo, and boosts as significant predictors of winPlacePerc.



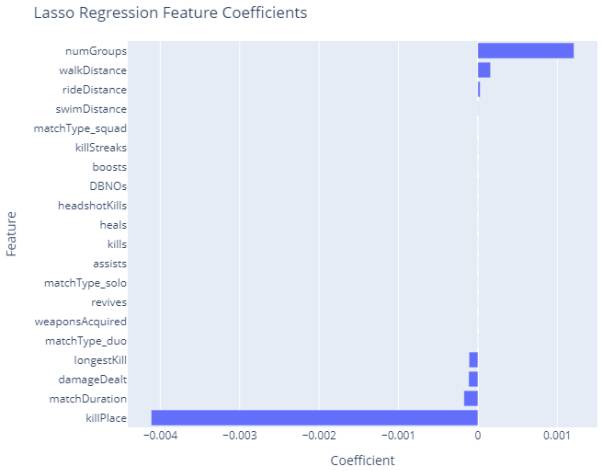
Ridge Regression Model:

- Achieved an MSE of 0.014833 on the validation set, similar to the linear regression model.
- Highlighted the importance of features like boosts, assists, and revives in predicting player performance.



Lasso Regression Model:

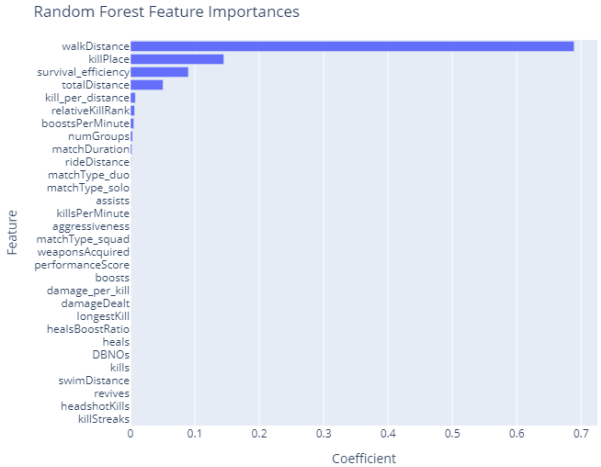
- Achieved an MSE of 0.020044 on the validation set, slightly higher than the linear and ridge regression models.
- Emphasized the significance of numGroups, walkDistance, and rideDistance in determining winPlacePerc.



B. Model Performance with Added Features

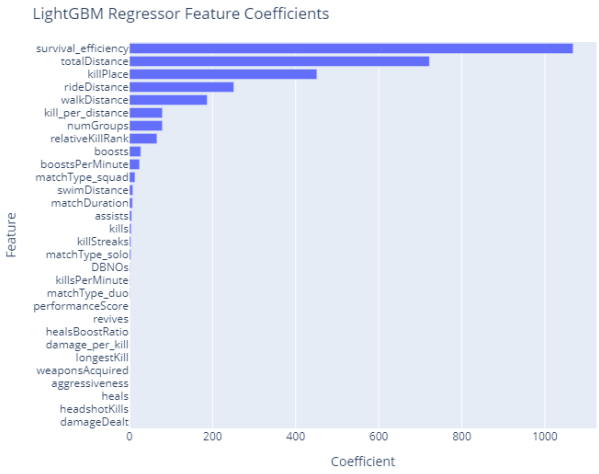
Random Forest Regression Model:

- Achieved an MSE of 0.000052 on the validation set, outperforming all other models.
- Identified walkDistance, KillPlace, and survival_efficiency as the most important features for prediction.



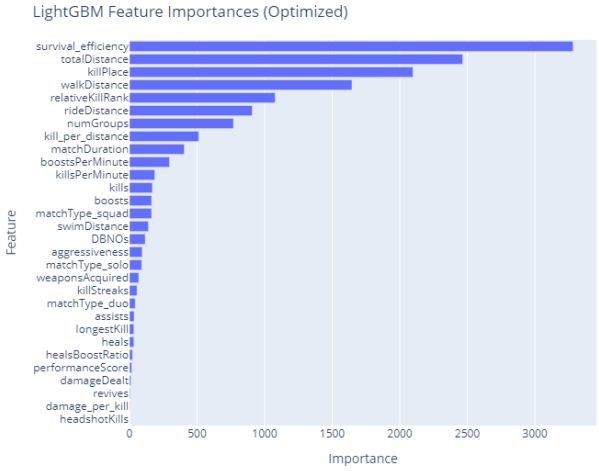
LightGBM Regression Model:

- Achieved an MSE of 0.000596 on the validation set, significantly lower than linear models.
- Emphasized the importance of survival_efficiency, totalDistance, and KillPlace in predicting player performance.



LightGBM Regression Model with Optuna Hyperparameter Tuning:

- Achieved an MSE of 0.000278 on the validation set, further improving predictive accuracy.
- Improved about 2 times the performance of the base LightGBM model.
- Confirmed the significance of survival_efficiency, totalDistance, and KillPlace in determining winPlacePerc.



C. Feature Impact

Important Features: walkDistance, KillPlace, survival_efficiency, and totalDistance were consistently identified as key predictors of winPlacePerc across all models.

Effective Representation: These features capture player activity, combat performance, and resource acquisition, reflecting the diverse strategies and skills required to succeed in PUBG.

D. Model Performance Analysis

Random Forest vs LightGBM Performance:

- Random Forest achieved superior MSE (0.000052) compared to base LightGBM (0.000596) due to:
  - Better handling of non-linear relationships and feature interactions through bagging
  - Natural resistance to overfitting through ensemble mechanism
  - Robust performance with default parameters
  - Enhanced noise tolerance through averaging predictions
- However, LightGBM advantages include:
  - Superior computational efficiency on large-scale datasets
  - Better scalability for production environments
  - Potential for further improvement through hyperparameter optimization

Model Selection Trade-offs: While Random Forest showed better raw performance metrics (MSE 0.000052), its lengthy training time (8 minutes 44 seconds) compared to LightGBM's swift execution (6 seconds) makes LightGBM more practical for large-scale deployment scenarios. This significant speed difference, combined with LightGBM's computational efficiency and optimization potential, justified its selection as the primary model despite marginally lower accuracy.

E. Interpretation of Model Parameters

Ensemble models highlighted the nonlinear influence of features, emphasizing the complexity of predicting game outcomes.

The importance of movement-related features suggests that strategic positioning and mobility are crucial for success in PUBG.

F. Conclusions

Success Factors: Ensemble models like Random Forest and LightGBM are more effective due to their ability to handle complex interactions within the data.

Model Selection Justification: LightGBM provided the best balance between performance and computational efficiency, making it suitable for large-scale datasets.

Failed Attempts: Linear models were inadequate in capturing the intricacies of player behavior, leading to higher prediction errors.

Recommendations: Future work could explore deeper hyperparameter tuning, incorporate additional data sources, or use more advanced techniques like deep learning for further improvements.

REFERENCES

[1] PUBG Finish Placement Prediction, "Competition Overview," Kaggle. Available: https://www.kaggle.com/c/pubg-finish-placement-prediction/overview. [Accessed: Nov. 19, 2024].

[2] Chocozzz, "How to Save Time and Memory with Big Datasets," Kaggle. Available: https://www.kaggle.com/code/chocozzz/how-to-save-time-and-memory-with-big-datasets/notebook. [Accessed: Nov. 20, 2024].