KAGGLE-PUBG-placement

D17-REPORT

TEAM:

Oskar Randmer

Gustav Nikopensius

Project repository: https://github.com/AlcadoSs/IDSPUBGFinishPlacementPrediction

Kaggle link: https://www.kaggle.com/c/pubg-finish-placement-prediction/overview

Business understanding

Our project does not have specific business aspect more like data science side to compare different ways in data preprocessing, different models and methods to reach lowest possible mean squared error (MSE) between predicted players placement and actual placement.

1. Identifying your project goals

a. Background (Game description)

PlayerUnknown's Battlegrounds (PUBG) is an online multiplayer battle royale game developed and published by PUBG Corporation. Up to 100 players compete to be the last man or team standing on different maps. The game starts by players jumping out of a plane without any gear and parachuting down to the ground, where the players can search buildings, ghost towns and other sites to find weapons, vehicles, armor, and other equipment to compete against other players or/and teams. In every few minutes size of playable map reduces towards a random location to increase the chances of encounters between the players and reduce players count.

b. Project and data-mining goals
   i. Find most important factors, what defines player placement in final ranking.
   ii. Data preprocessing- data cleaning, normalization, transformation and feature extraction and selection.
   iii. Compare MSE of Simple linear regression, Ridge regression, Lasso regression, Random Forest and Lightgbm.

c. Project and data-mining success criteria
    i. Reach below 0.05 MSE.

2. Assessing your situation

a. Inventory of resources
    i. 2 students with laptops and PCs. Python and Jupyter Notebook.

b. Requirements, assumptions, and constraints
    i. Finish project according to the project plan.
    ii. Achieve set goals.

c. Risks and contingencies
    i. Lack of knowledge in data science. Ask from instructors or try to find answer from Internet.
    ii. Teamwork is not so good as expected. Communicate with teammate and it is not working out finish project by yourself.
    iii. Problems completing set task. Communicate with teammate, maybe switch task for other with teammate.

d. Terminology
    i. **Simple linear regression** - sklearn.linear_model.LinearRegression
    ii. **Ridge regression** - sklearn.linear_model.Ridge
    iii. **Lasso regression** - sklearn.linear_model.Lasso
    iv. **Random Forest** - sklearn.ensemble RandomForestRegressor
    v. **Lightgbm (Light Gradient Boosting Machine)** – LGBMClassifier
    vi. **mean squared error (MSE)** - The mean squared error tells you how close a regression line is to a set of points. It does this by taking the distances from the points to the regression line (these distances are the "errors") and squaring them. The squaring is necessary to remove any negative signs. It also gives more weight to larger differences. It is called the mean squared error as you are finding the average of a set of errors.

Data understanding

1. Gathering data

a. The data is from the Kaggle competition PUBG Finish Placement Prediction.

2. Describing data

a. The data is composed of two csv files: train_V2.csv, test_V2.csv. Both have 28 features to predict from, and the train data also contains the target feature.
b. We have player data from 65,000 PUBG games. A row of in-game statistics and match details for each player in each game: the train data contains over 4 million such rows.

3. Exploring data

a. The target feature is a percentile winning placement: 1 corresponds to 1st place, 0 corresponds to last. Most of the data to predict from is a numeric representation of what each player did during the game: number of kills, distance run, teammates assisted, number of weapons acquired etc.
b. The game can be played in solo, duo or squad (4 players) mode. The weight of some features is tied to the game mode, for example the number of teammates revived, which will always be 0 in a solo game, but might be an important statistic in predicting victory or defeat in a team game.
c. There are three ranking features: rankPoints, killPoints, winPoints. For each row it is either rankPoints == -1 and killPoints >0 and winPoints >0, or rankPoints > 0 and killPoints == winPoints == 0. This means if we are going to want to use these features, we are going to have to find a way to create one feature out of these three.
d. There are some strong correlations between the target feature and in-game statistics, in particular the distance walked, weapons acquired, boost items used and killplace which is the end of game ranking according to number of kills during the match. Looking at only solo games the features associated with kills and dealt damage become even more important.

4. Verifying data quality

a. The data is good for use. Some feature engineering is going to be necessary concerning game modes and player rankings.

Project plan

1. **Plan**
    1. **Data visualization - Oskar (6h)**
    2. **Data preprocessing - Gustav (15h)**
        i. Data cleaning
        ii. Data normalization
        iii. Data transformation
    3. **Simple linear regression, Ridge regression, Lasso regression, Random Forest, Lightgbm – Oskar (15h)**
        i. Models building and evaluation - parameters tuning and feature selection
    4. **Summary of results and visualization - Gustav (6h)**
    5. **Video/Poster – Oskar and Gustav (8h each)**

2. **List the methods and tools**
    1. **Pandas** and **NumPy** for data preprocessing.
    2. **Matplotlib** and **Seaborn** for data visualization and final summary.
    3. **Keras**, **scikit-learn** and **Lightgbm** for models building and evaluation
    4. **Sony Vegas**, **Photoshop**, some recording software for video/poster