Course: Statistics for Data Science

Group Project:

Analysis of the contributing factors and prediction of the outcome of a chess game

Submitted on: 12/12/2022

Group 1

By:

Arvind Shankar
Atiyeh Monfared
Raghupathy Kolandavelu
Mantajvir Singh

1- Introduction

Predicting the result of a game is always an exciting task. In this report, we use a dataset of the chess games from Kaggle which includes the attributes of the players (e.g. their level of proficiency), the attributes of the game (e.g. Termination) and the result of the game (i.e. win/ lose) to study the possible impact of these factors on the outcome of the game. We suggest three hypotheses and test them. In the next step, we built a prediction model using Logistic regression. The reason we chose Logistic regression is that the dependent variable in our model (i.e. the outcome of the game) is binary with two possible values of Win or Lose.

2- Objective

Our goal in this project is to study the factors that affect the chance of winning in a chess game. The dependent variable is the "Winner" and we consider possible contributing factors such as the Elo of the players, number of moves, etc. We also investigate the correlation that might exist between the independent variables.

Hypothesis to Test:
Hypothesis 1: White Player has a greater chance of winning than the Black player.

Hypothesis 2: The greater the difference in ratings of two players, the higher the probability of the higher-rated player to win.

Hypothesis 3: For games in the bullet (Less than the 179 seconds) categories, the white player is more likely to win.

Predictive Model:
Predict the outcome of a game based on the information we have about the two players and other attributes of the game.

3- Methodology

In this project, we first explored the dataset to assess its quality and define the steps required for data cleaning and data preparation phase. We checked for duplicates and missing data and deleted the columns such as names of the players which were not helpful to our analysis. In the analysis phase, we tested the hypothesis and also developed a predictive model. Knowing that our dependent variable, win color, is a categorical variable with two values, black or white, we should use the Logit regression model for prediction. Details of each phase are explained in the following sections.

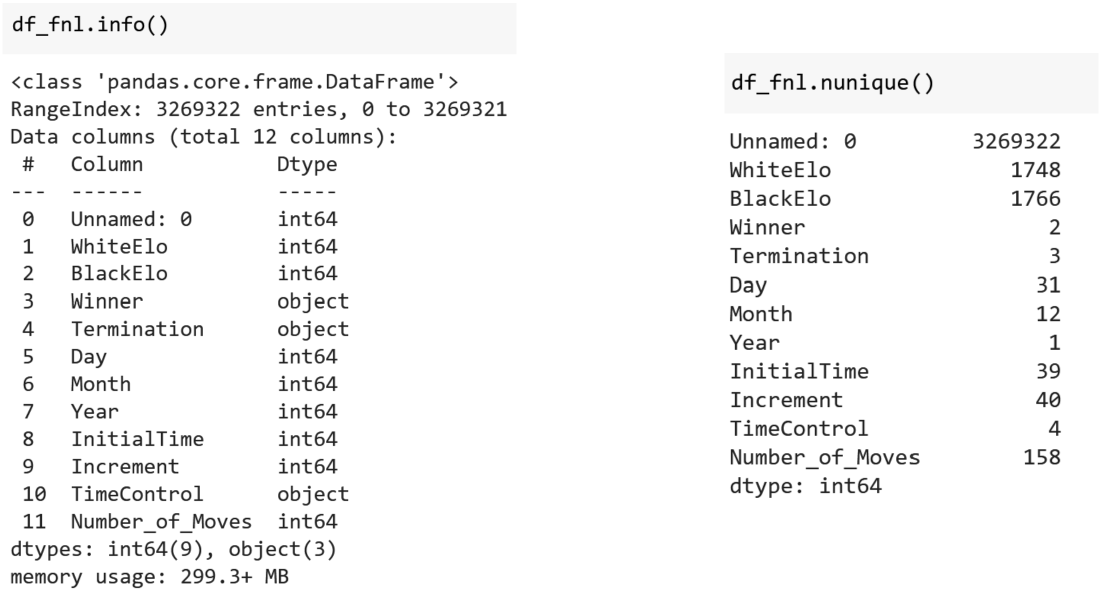
4- Data Exploration and Preparation

In this report, we use the 15 Million Chess Games from Lichess (2013-2014) dataset shared on Kaggle. This dataset contains data from all rated games (around 15.000.000) played in Lichess website from January 2013 to December 2014.

Features of the dataset are listed below:

- WhiteElo: Elo of the player with white pieces which shows their level of proficiency
- BlackElo: Elo of the player with black pieces which shows their level of proficiency
- WhiteName: Name of the player with white pieces
- BlackName: Name of the player with black pieces
- Winner: Color of the winning pieces. If the game ended in Draw it shows it.
- Termination: How the game ended, it can be: Normal, Time Forfeit, Abandon or Rules infraction.
  - Normal: When the game end in checkmate, abandon or draw
  - Time Forfeit: When one of the players runs out of time
  - Abandon: When in a competition one of the players doesn't make a move
  - Rules infraction: When one of the players is banned
- Site: URL of the game
- Day: Day when the game was played
- Month: Month when the game was played
- Year: Year when the game was played
- InitialTime: Time each player has before starting the game in seconds
- Increment: Increment in the time after each player makes a move in seconds
- TimeControl: Classification of the games based on the estimated duration of a game calculated as InitialTime+ 40*Increment. If estimated duration:
  - <=29s: Ultrabullet
  - <=179s: Bullet
  - <=479s: Blitz
  - <=1499s: Rapid
  - Bigger or equal than 1500s: Classical
- Opening: Opening Name
- ECO: Classification of the games based on the ECO(Encyclopaedia of Chess Openings) code
- Number of Moves: Number of moves of the game
- FEN: FEN of the game

To explore the dataset, we used the info and nunique codes and the results are presented in Table1.



```
df_fnl.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3269322 entries, 0 to 3269321
Data columns (total 12 columns):
 #   Column           Dtype 
---  ------           ----- 
 0   Unnamed: 0       int64 
 1   WhiteElo         int64 
 2   BlackElo         int64 
 3   Winner           object
 4   Termination      object
 5   Day              int64 
 6   Month            int64 
 7   Year             int64 
 8   InitialTime      int64 
 9   Increment        int64 
 10  TimeControl      object
 11  Number_of_Moves  int64 
dtypes: int64(9), object(3)
memory usage: 299.3+ MB
```

```
df_fnl.nunique()
```

```
Unnamed: 0         3269322
WhiteElo              1748
BlackElo              1766
Winner                   2
Termination              3
Day                     31
Month                   12
Year                     1
InitialTime             39
Increment               40
TimeControl              4
Number_of_Moves        158
dtype: int64
```

Table 1- Dataset exploration (left: column information, right: number of unique values in each column)

The dataset was so big originally with about 15 million records which made the processing time longer than usual so we decided to use the data from 2013 instead of the whole dataset. This step reduced the volume of the data to about 3 million which accelerated the processing time of the codes in Python.

Moreover, as a part of the initial data exploration and cleanup some insignificant features such as the Name of the players, link to the game log, day, month, year, initial time, increment, opening name, FEN were dropped from the dataset.

There were 494 unique values in the ECO column and 2990 for the opening name which were not interpretable so we dropped these variables as well.

We checked for duplicates and missing data and there were none in the dataset.
We created dummies for categorical variables “Termination” and “Time control” and also transformed the values in the “Winner” column which is the dependent variable of our model to 0 and 1 to enable us to perform our regression analysis.

5 - Data Visualization

5-1- Continuous Independent (Predictor) Variables

Three of the independent variables in our model are continuous and their distributions are presented here.

The distribution of the level of proficiency of the two players is illustrated in Figure 1.

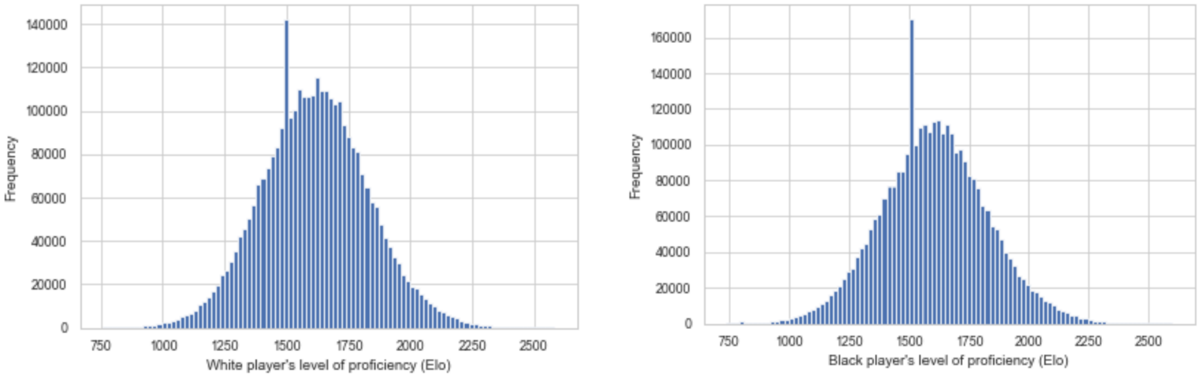


Figure 1- Distribution of players' rating (left: White player, right: Black player)

The bell shape of the distribution is very similar to normal distribution. Mean rate for the black and white players are 1602 and 1612 respectively.

The distribution of the number of moves in the games is depicted in Figure 2. Minimum, maximum and average number of moves in this dataset are 1, 191 and 33 respectively. Discounting the outliers on the right, the number of moves follow a normal distribution

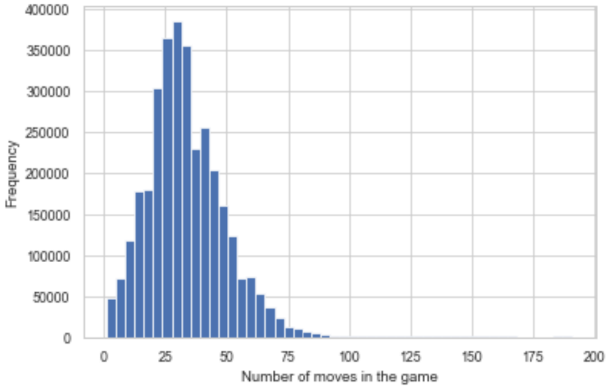


Figure 2-Distribution of number of moves

5-2- Categorical Independent (Predictor) Variables

Number of observations for each category of column "Termination" is presented in Figure 3.

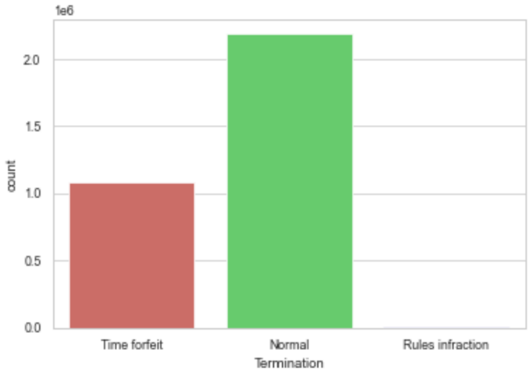


Figure 3- Counts for each category of Termination

Number of observations for each category of column "Time Control" is presented in Figure 4.

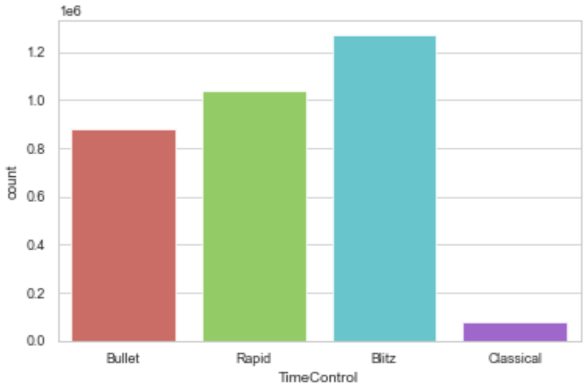


Figure 4- Counts for each category of Time Control

5-3- Dependent (Response) Variable

Number of observations when white or black player won the game is presented in Figure 5.

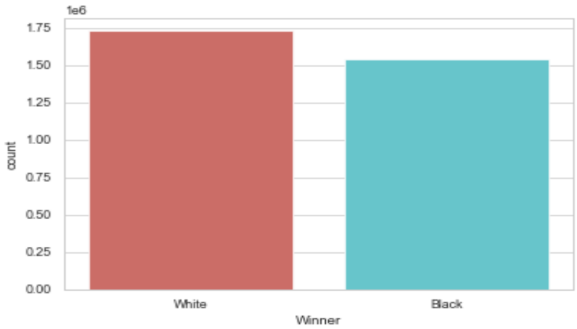


Figure 5- Counts for each colour winning the game

6 - Analysis

6-1- Hypothesis Testing

Hypothesis 1: White Player has a greater chance of winning than the Black player.

There are two players in the game ; White (one with white pieces) and Black (player with black pieces). As per the rules of the game, the player with the white pieces always starts the game. We believe the white player has an edge over the black player as white always has a head start in the game , we propose the statement that "White players have higher chances of winning than Black players".

To test our hypothesis, we took 1000 samples with a sample size of 100. If both White and Black have an equal probability to win the game, the mean of the sampling distributions should be 0.5 To validate our hypothesis, the following codeblock was executed and a plot was done to visualize the outcome:

```
samplesize = 100
whitewin = 0
x_data = []
y_data = []

for num_simulations in range(1, 1001):
    df_sample = df_fnl.sample(n=samplesize)
    whitewin += len(df_sample[df_sample.Win_Color == 1])
    y_data.append(float(whitewin) / float(num_simulations*samplesize))
    x_data.append(num_simulations)

plt.plot(x_data,y_data)
```

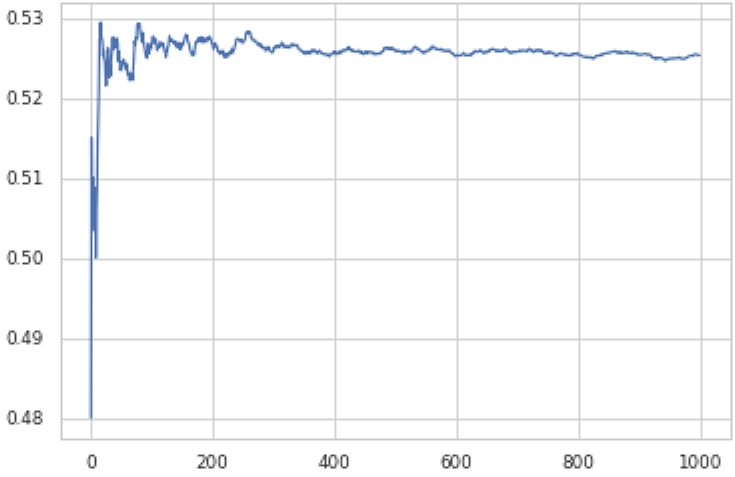


Figure 6 - Convergence of the samples' mean value

From the above visualization, we were able to infer that by computing the percentage of wins from the various samples, the probability of white player winning is 0.53, greater than the expected probability of 0.50.

But, is this good enough to make predictions with a 95% confidence level just based on the color? The answer for this question is explored in the following section.

Definition of Hypothesis:
H0 : Both Black and White players have an equal chance to win
HA: Both Black and White players have do not have an equal chance to win

Point Estimate: 0.52 **(Derived from a sample of 1000 game sets)**
Standard Error: 0.05

Distance of the sample mean from the NULL hypothesis mean
Z Value: 0.5064000000000002

Area covered under the curve
New P-value: 0.6125758745724573

As we see that the new p-value is greater than 0.05 (5%) and hence the NULL hypothesis cannot be rejected. There is no strong evidence to reject that winning is equal for both white and black.

The NULL hypothesis can also be explained by following graph (Scaled to a standard deviation of 0.5):

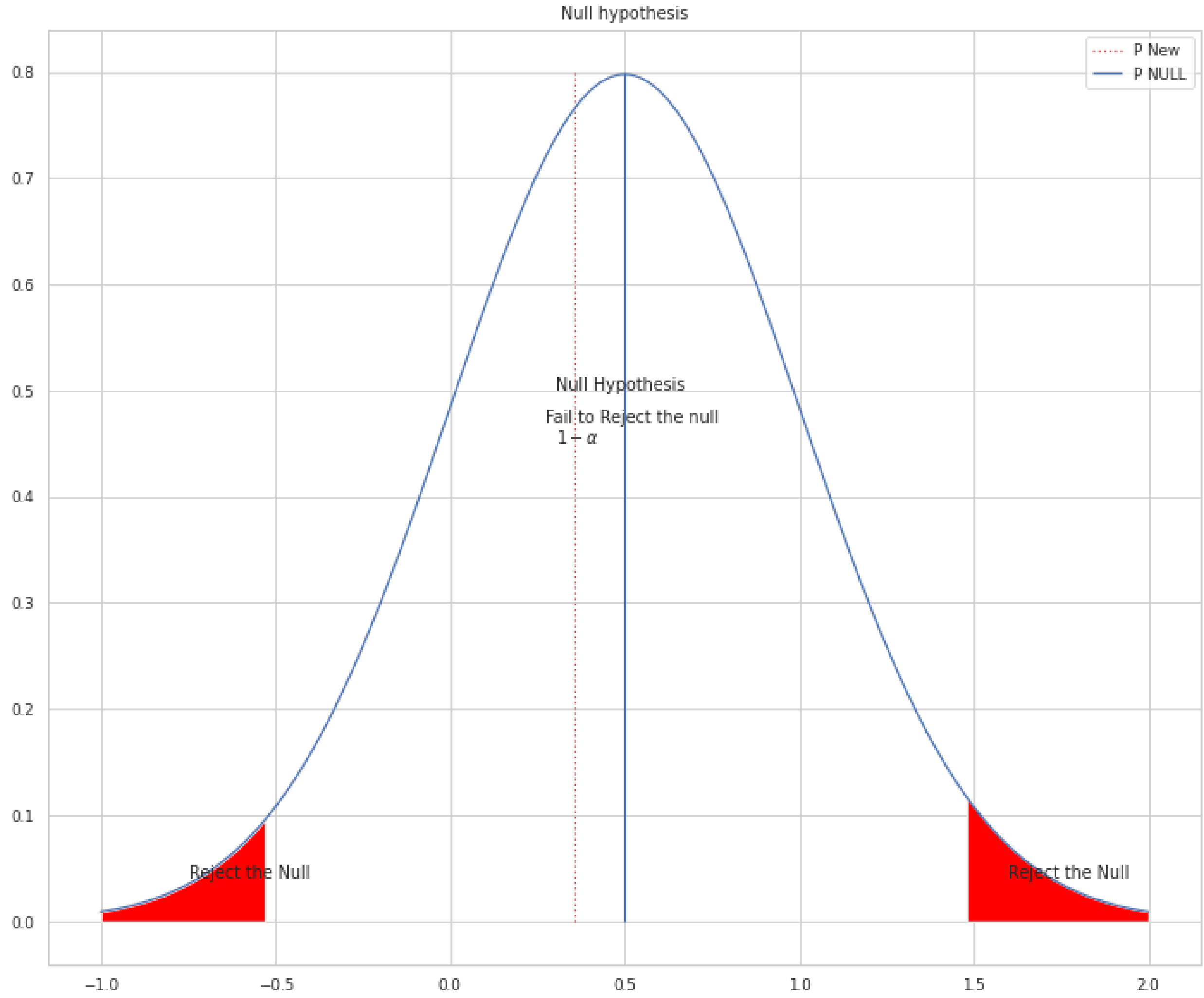


Figure 7- Hypothesis testing visualization

Hypothesis 2: The greater the difference in ratings of two players, the higher the probability of the higher-rated player to win.

As we saw that some players are highly rated as compared to other players, we decided to evaluate if the rating of the player affects the winning probability for the player.

Value 1 : When a high rated player wins the game

Value 0: When a lower rated player wins the game

Definition of Hypothesis:
H0 : Both players have equal chance of winning the game
HA: One of the players has a higher chance to win

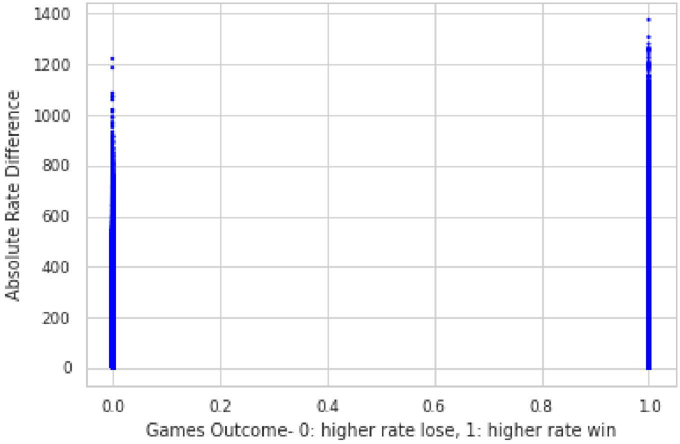


Figure 8- Scatter plot of the rate difference vs game outcome

As observed from the above scatter plot, in most of the games when the rating difference of players is more than 1000 points, the higher rated player wins (see the dots piled on value 1 in x axis) . Splitting the data frame based on the rating of players, we found that the higher rated player won **2.1M** games out of **3.25M** games while the lower rated player won in **1.1M** games. Distribution of the rating difference for the two cases when a higher rated player wins or the lower rated player wins is depicted in below illustration.

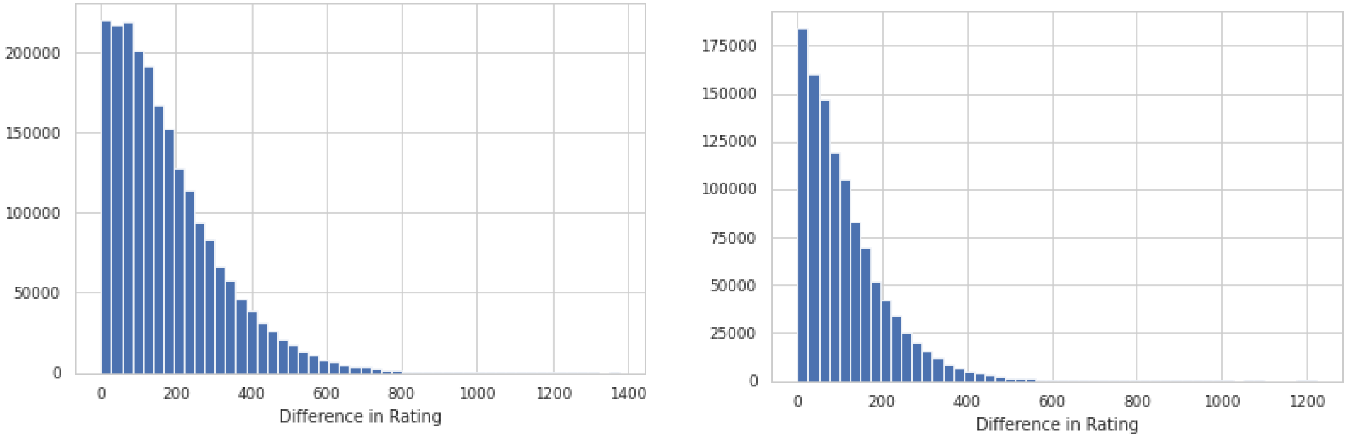


Figure 9- Distribution of rating difference (left: higher rated player wins, right: lowe rated player wins)

We decided to further refine this chart by restricting the data frame where the difference in rating between the players is < 1000 points.

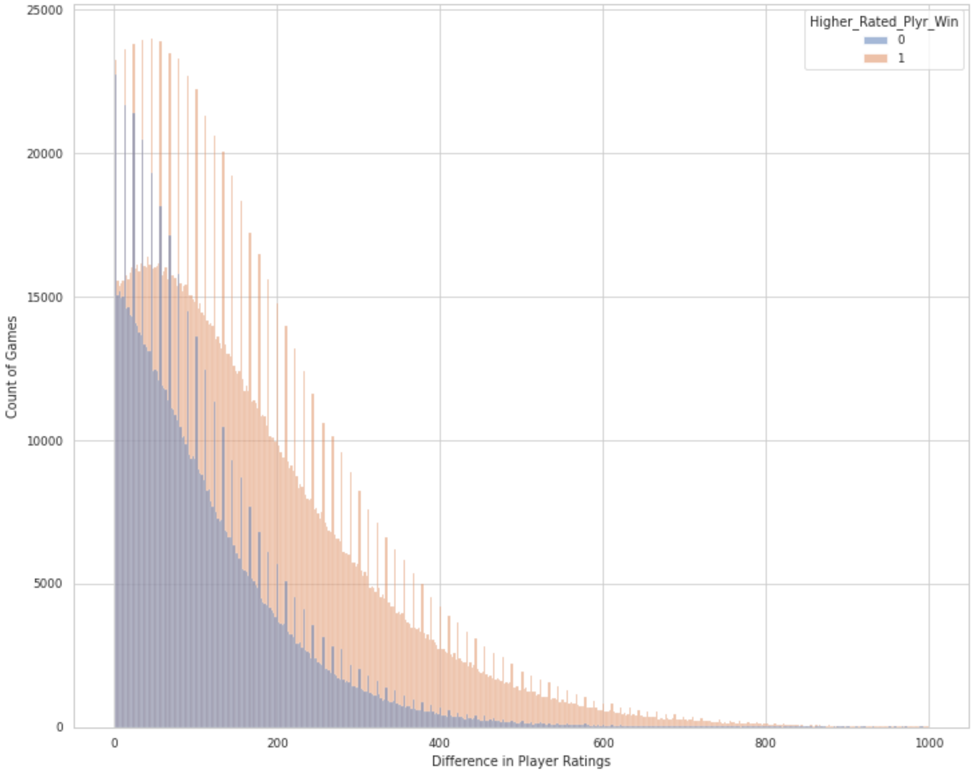


Figure 10- Rate difference distribution and comparison

Based on the above graph, it is evident that as the Difference in Rating between the players increases, the Higher Rated Player wins more games than the lower rated player. And from the computation of probability_higher_rating_win column, the probability is 66% which is ~2x times than the probability of win of a lower rated player.

Hypothesis 3: For games in the Bullet (Less than the 179 seconds or 3minutes) categories, the white player is more likely to win.

As the game has been divided into different categories based on time, if the game is under 3 minutes, it is classified as Bullet and if it is more than 3 minutes then it is classified as Blitz or Rapid. As per hypothesis, our main focus was on Bullet and White Player. To find the probability, we took 1000 random samples with a sample size of 100. The plot below visualizes the probability of white win and convergence probability.

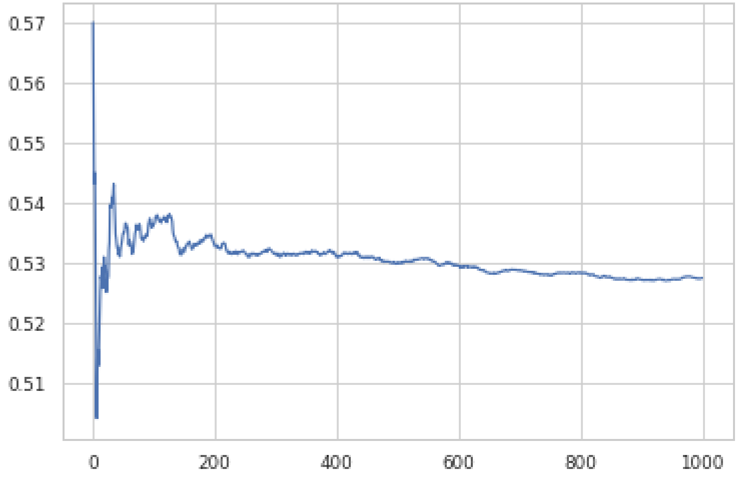


Figure 11: Probability of Whiteplayer win under Bullet categories

The likelihood of white winning in any game (less than three minutes, Gametype = ” Bullet” was utilized for this illustration) is approximately 0.53 according to the plot above. Although this is marginally above the expected population mean of 0.5, it is not significant enough to suggest a bias in favor of white for any game.

Definition of Hypothesis:
H0 : Both Black and White players have an equal chance to win in Bullet Games
HA: Both Black and White players have do not have an equal chance to win in Bullet Games

To test the Null hypothesis, we find the following values:

Point estimate: 0.52 ((Derived from a sample of 1000 game sets)
Standard Error: 0.05

Distance of the sample mean from the NULL hypothesis mean
Z Value: 0.55

Area covered under the curve
New P-value: 0.5813595006947428

With 95% confidence, the NULL hypothesis cannot be rejected because the new p-value is greater than 0.05. There is no strong evidence to reject that winning is equal for both White and black in Bullet games.

The NULL hypothesis can also be explained by following graph (Scaled to a standard deviation of 0.5):

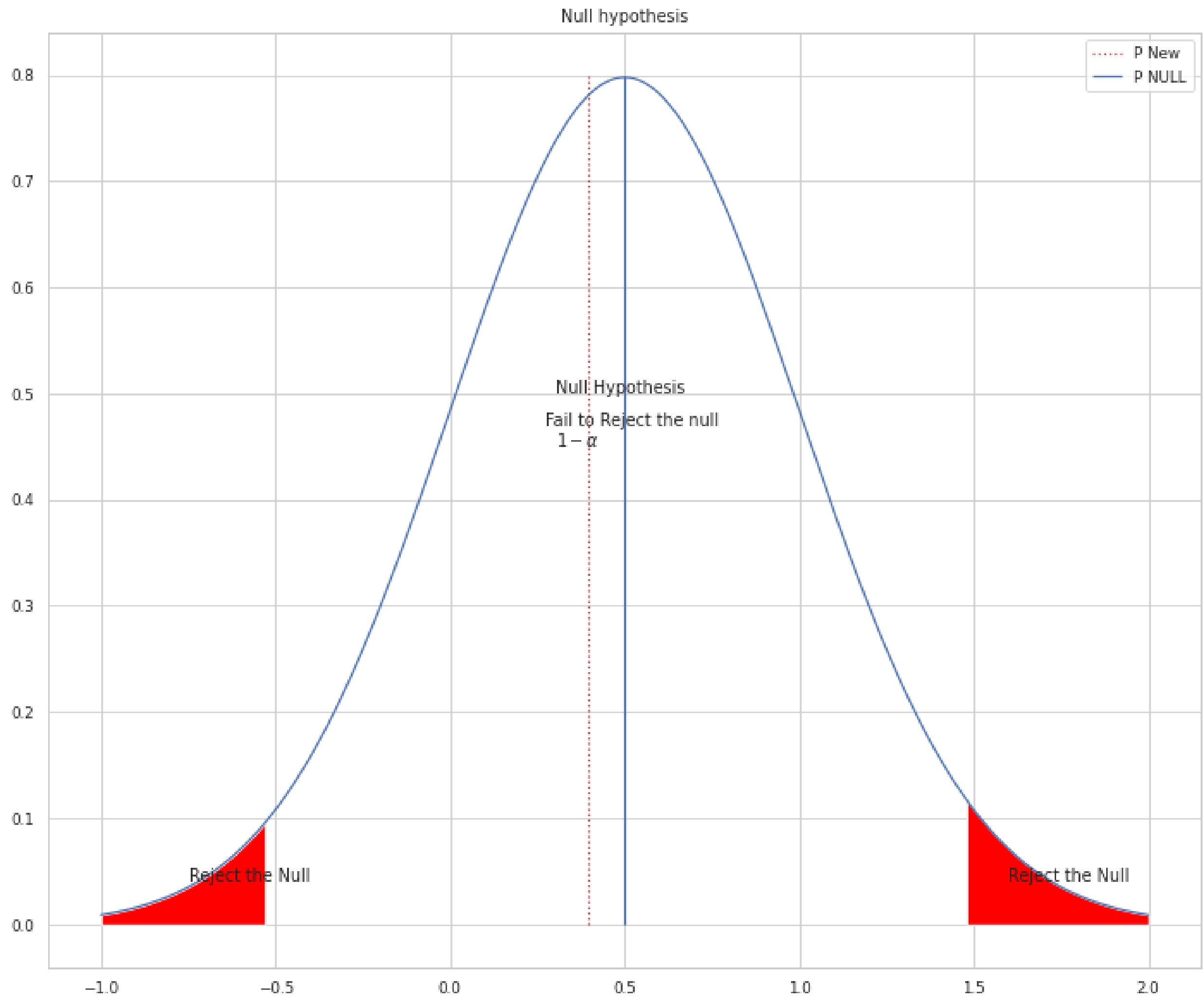


Figure 12 : Null Hypothesis

6-2- Predictive Model

As explained in the previous sections, we used Logistic Regression for our predictive model.

Features from the dataset that were selected to be used in the predictive model are:

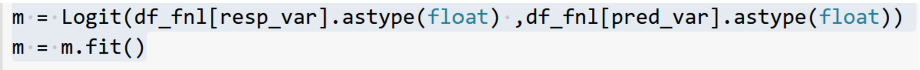
Predictor Variables

- Diff_Rating: absolute difference in ratings of the two players
- Number_of_Moves: total number of moves in the game
- TimeControl_Blitz: dummy variable for time control type Blitz
- TimeControl_Bullet: dummy variable for time control type Bullet
- TimeControl_Classical: dummy variable for time control type Classical
- TimeControl_Rapid: dummy variable for time control type Rapiz
- Term_Rul_Normal: dummy variable for term rule Normal

- Term_Rul_Infraction: dummy variable for term rule Infraction
- Term_Rul_Time_Forfeit: dummy variable for term rule Forfeit
- Rating_B: dummy variable when the rating of Black player is higher
- Rating_EQ: dummy variable when the rating of the two players is equal
- Rating_W: dummy variable when the rating of White player is higher

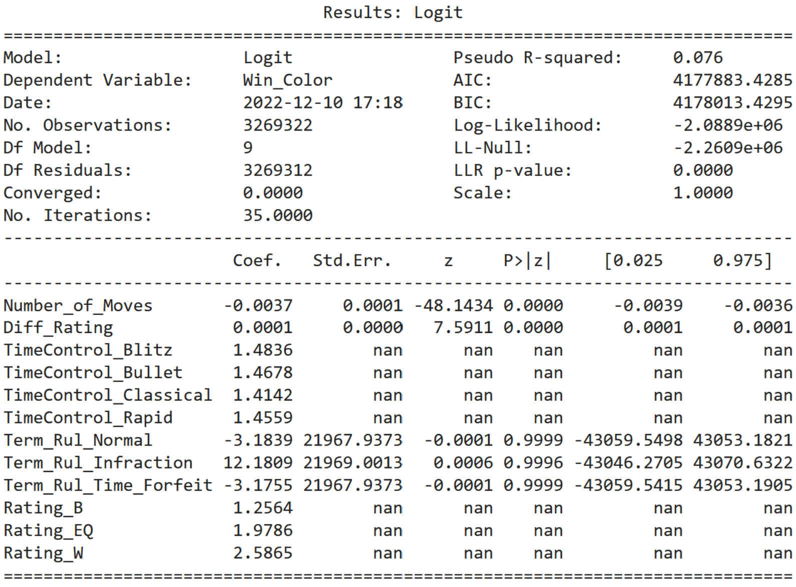
Response Variable: Win_Color

We initially ran a regression model by leveraging the Logit function:



```
m = Logit(df_fnl[resp_var].astype(float) ,df_fnl[pred_var].astype(float))
m = m.fit()
```

The summary is presented in Table 2.

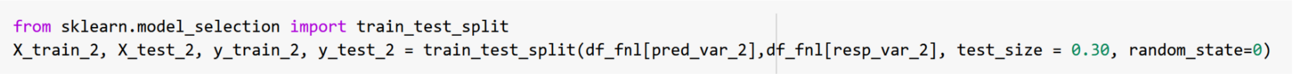


```
                             Results: Logit
=================================================================================
Model:                  Logit                 Pseudo R-squared:     0.076
Dependent Variable:     Win_Color             AIC:                  4177883.4285
Date:                   2022-12-10 17:18      BIC:                  4178013.4295
No. Observations:       3269322               Log-Likelihood:       -2.0889e+06
Df Model:               9                     LL-Null:              -2.2609e+06
Df Residuals:           3269312               LLR p-value:          0.0000
Converged:              0.0000                Scale:                1.0000
No. Iterations:         35.0000
---------------------------------------------------------------------------------
                       Coef.   Std.Err.     z     P>|z|     [0.025      0.975]
---------------------------------------------------------------------------------
Number_of_Moves       -0.0037     0.0001 -48.1434 0.0000     -0.0039     -0.0036
Diff_Rating            0.0001     0.0000   7.5911 0.0000      0.0001      0.0001
TimeControl_Blitz      1.4836        nan      nan    nan         nan         nan
TimeControl_Bullet     1.4678        nan      nan    nan         nan         nan
TimeControl_Classical  1.4142        nan      nan    nan         nan         nan
TimeControl_Rapid      1.4559        nan      nan    nan         nan         nan
Term_Rul_Normal       -3.1839 21967.9373  -0.0001 0.9999 -43059.5498 43053.1821
Term_Rul_Infraction   12.1809 21969.0013   0.0006 0.9996 -43046.2705 43070.6322
Term_Rul_Time_Forfeit -3.1755 21967.9373  -0.0001 0.9999 -43059.5415 43053.1905
Rating_B               1.2564        nan      nan    nan         nan         nan
Rating_EQ              1.9786        nan      nan    nan         nan         nan
Rating_W               2.5865        nan      nan    nan         nan         nan
=================================================================================
```

Table 2- Logit regression summary report

As it can be inferred from the summary statistics, "Term_Rul_" variables have extremely high p values which indicates that the impact of this feature is not significant to the response variable. Therefore, this variable was removed.

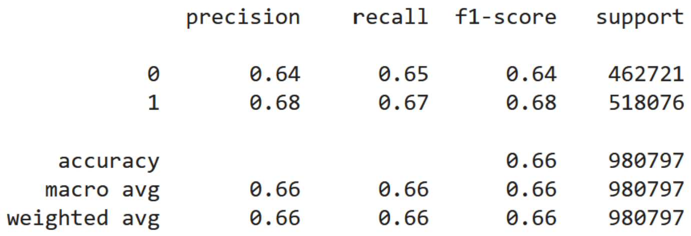
We then used the imported the sklearn library and trained the model with 70% of the dataset and tested the model prediction using the test dataset (30%)



```
from sklearn.model_selection import train_test_split
X_train_2, X_test_2, y_train_2, y_test_2 = train_test_split(df_fnl[pred_var_2],df_fnl[resp_var_2], test_size = 0.30, random_state=0)
```

We also performed feature scaling before fitting the model. Finally we ran the model evaluation report to retrieve the various performance characteristics of the model.

Model evaluation report is presented in the below table.



```
              precision    recall  f1-score   support

           0       0.64      0.65      0.64    462721
           1       0.68      0.67      0.68    518076

    accuracy                           0.66    980797
   macro avg       0.66      0.66      0.66    980797
weighted avg       0.66      0.66      0.66    980797
```

Table 3- Precision, Recall, F1-score, Support

The model had an accuracy score of 66%, with a precision score of 68%, a True Positive Rate (Recall / Sensitivity) of 67% and specificity of 65%. Furthermore, the model also had a F1 Score (Harmonic Mean) of 68%.

As it was evident from the accuracy and the f1 scores, the model was able to predict the outcome of White Win (variable 1) with a reasonable accuracy.

Visualization on the confusion matrix is presented in the below figure for your reference.

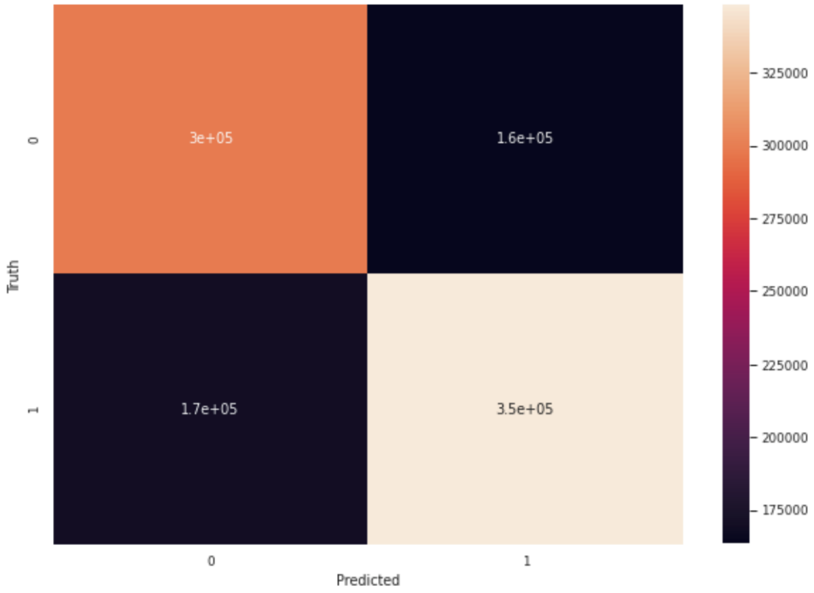


Figure 13 : Confusion Matrix

7 - Conclusion

In this project, we validated three hypotheses and built a predictive model to predict the outcome of a chess game based on a dataset from Kaggle.

- With regards to hypothesis 1, we could not find enough evidence to reject the hypothesis that the chance of winning is equal for both white and black.
- As far as hypothesis 2 is concerned, we found that in 66% of the games, the player with higher rating has won the game which is almost double the number of times a player with lower rank actually won the game (34%). Furthermore, as the difference in rating between the players increases, the higher rated player is likely to win more games than the lower rated player.
- For hypothesis 3, we rejected the null hypothesis which means that for games in the Bullet (Less than the 179 seconds or 3minutes) category, the white and black players are equally likely to win.
- According to our predictive model, number of moves and difference in rating are two impactful features that are significant to predict the outcome of the game.

References

15 Million Chess Games from Lichess (2013-2014). (n.d.). Retrieved December 9, 2022, https://www.kaggle.com/datasets/maca11/chess-games-from-lichess-20132014

OpenIntro Statistics (4th Ed.). https://leanpub.com/openintro-statistics