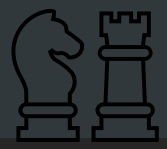
# Chess Winner Prediction

SALEM EID

2021 DEC 16

# Chess Winner Prediction

1. Introduction
2. Dataset
3. Tools to be used.
4. Processes
    - Cleaning
    - Preprocessing
    - EDA
    - Modelling
5. Results
6. Future



# Introduction

Imagine that we have a chess competition to prepare for and we need to find each player favorite opening move, to increase our chance of winning through data. The main goal of this data science research is to answer the following question:

Can I Predict the winner of a chess match using ML and past games dataset ?

# Dataset

The dataset was obtained from Kaggle, it's rich in information which needs to be handled with first before performing nay modellings it contains 20,000 datapoints and 16 features.

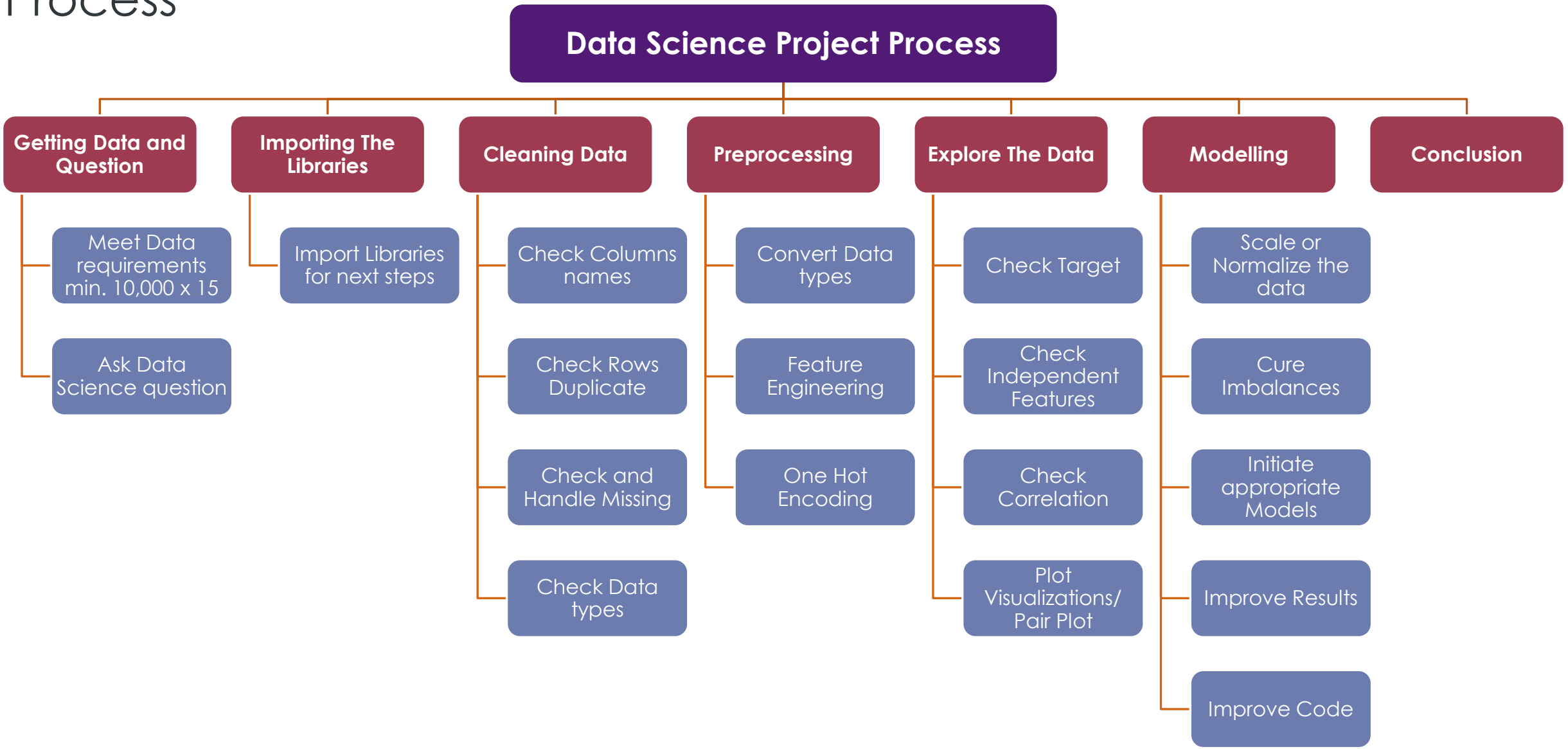
| Data Feature | Description |
|---|---|
| Game ID | System generated game ID. |
| Rated (T/F) | Either the game is rated for points or casual |
| Start Time | start time of the game |
| End Time | end time of the game |
| Number of Turns | number of turns the game took |
| Game Status | game end status (mate, out of time, draw, resign) |
| Winner | White, Black or Draw |
| Time Increment | A fixed amount of time (in minutes) assigned at move 0 for the entire game, and increment (in seconds) is the Amount added after each move., more info increment methods |
| White Player ID | White player name |
| White Player Rating | White player level - the higher the more victories and experience |
| Black Player ID | Black player name |
| Black Player Rating | Black player level - the higher the more victories and experience |
| Moves | All Moves in Standard Chess Notation; |
| Opening Eco | Standard Code for opening moves |
| Opening Name | A set of moves in the beginning of the game which form into a move |
| Opening Ply | Number of moves in the opening phase which belongs to the opening moves |

kaggle

# Tools to be used

| Programming Language | Data Handling Libraries | Modelling Libraries | Visualizations Libraries |
|---|---|---|---|
|  |  |  |  |
| |  |  |  |

# Process


Data Science Project Process
Getting Data and Question
Meet Data requirements min. 10,000 x 15
Ask Data Science question
Importing The Libraries
Import Libraries for next steps
Cleaning Data
Check Columns names
Check Rows Duplicate
Check and Handle Missing
Check Data types
Preprocessing
Convert Data types
Feature Engineering
One Hot Encoding
Explore The Data
Check Target
Check Independent Features
Check Correlation
Plot Visualizations/ Pair Plot
Modelling
Scale or Normalize the data
Cure Imbalances
Initiate appropriate Models
Improve Results
Improve Code
Conclusion

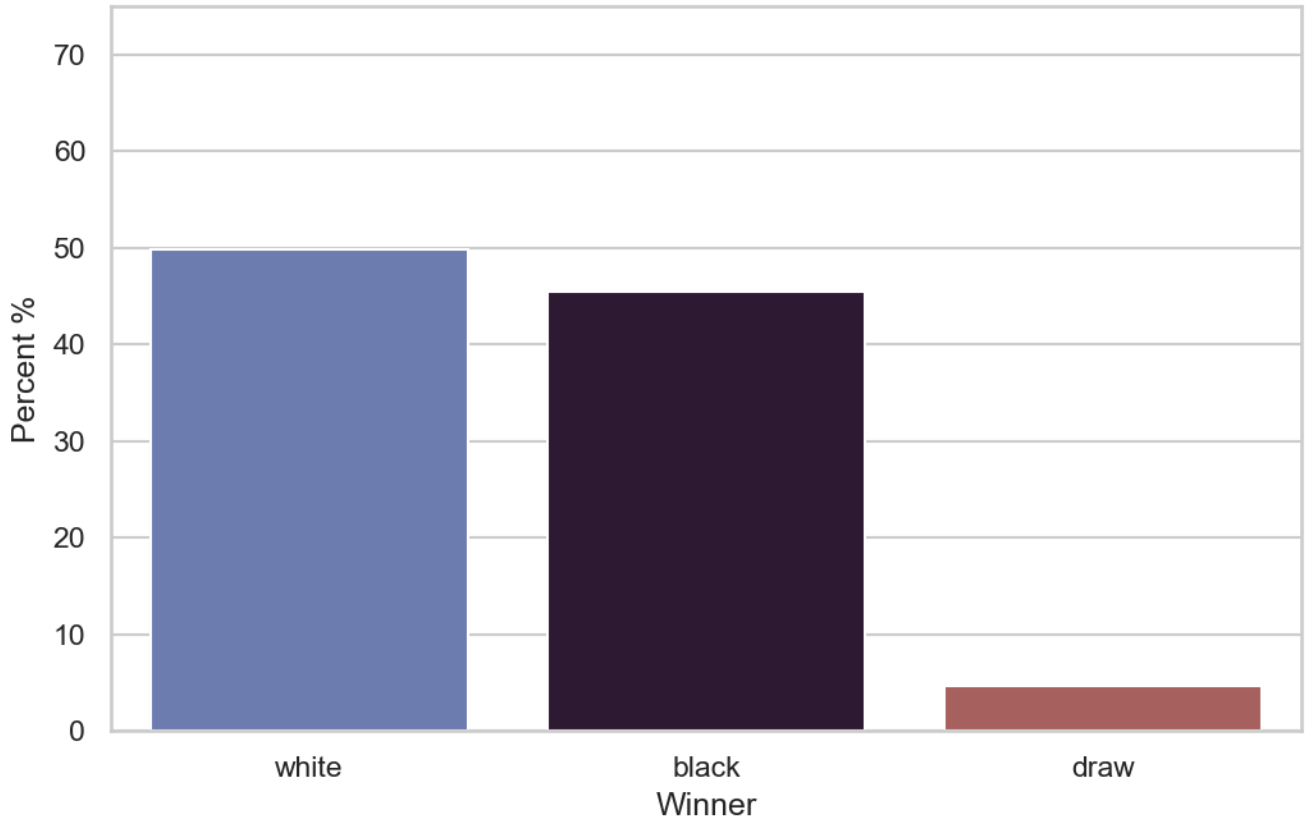
# Cleaning Data

- ✓ 400 + Duplicated data records
- ✓ Converted timestamp feature(created_at, last_move_at) which was in Unix format
- ✓ Decode increment_code column
- ✓ Shorten opening names which had many sub variations
- ✓ Extracted rating difference for each game
- ✓ Categorized games based on rating average.

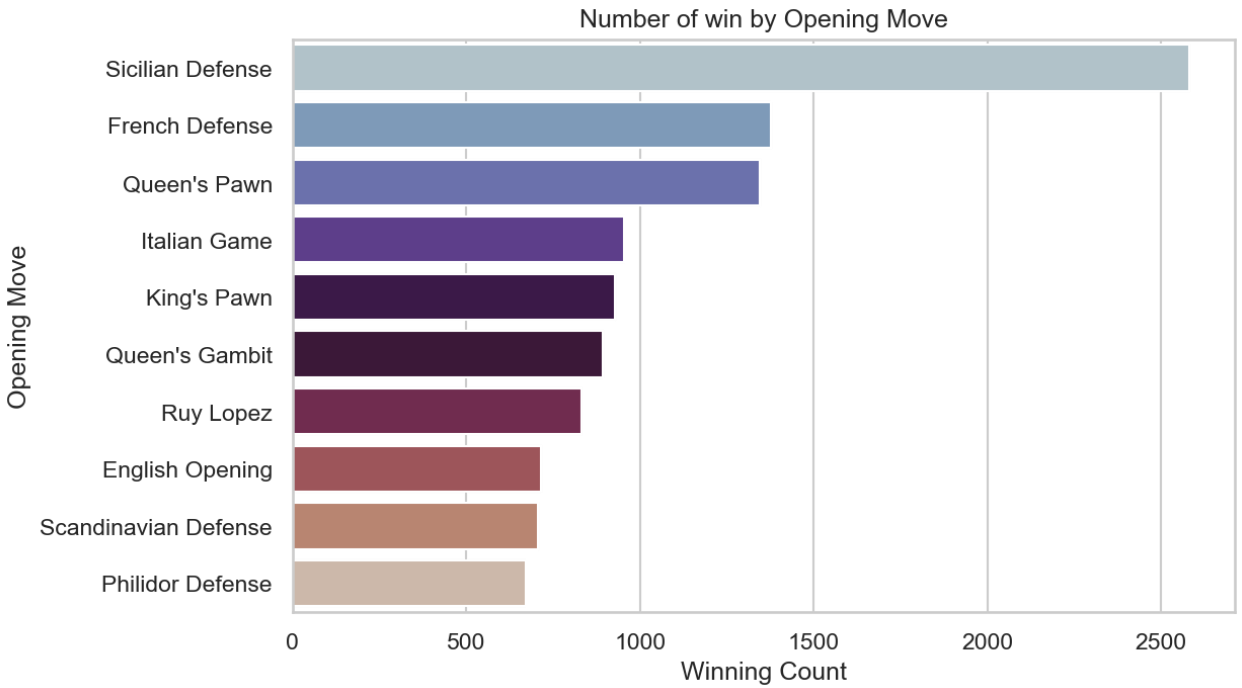
# Target data

The fight between white and black is a close call , white edges a little bit.

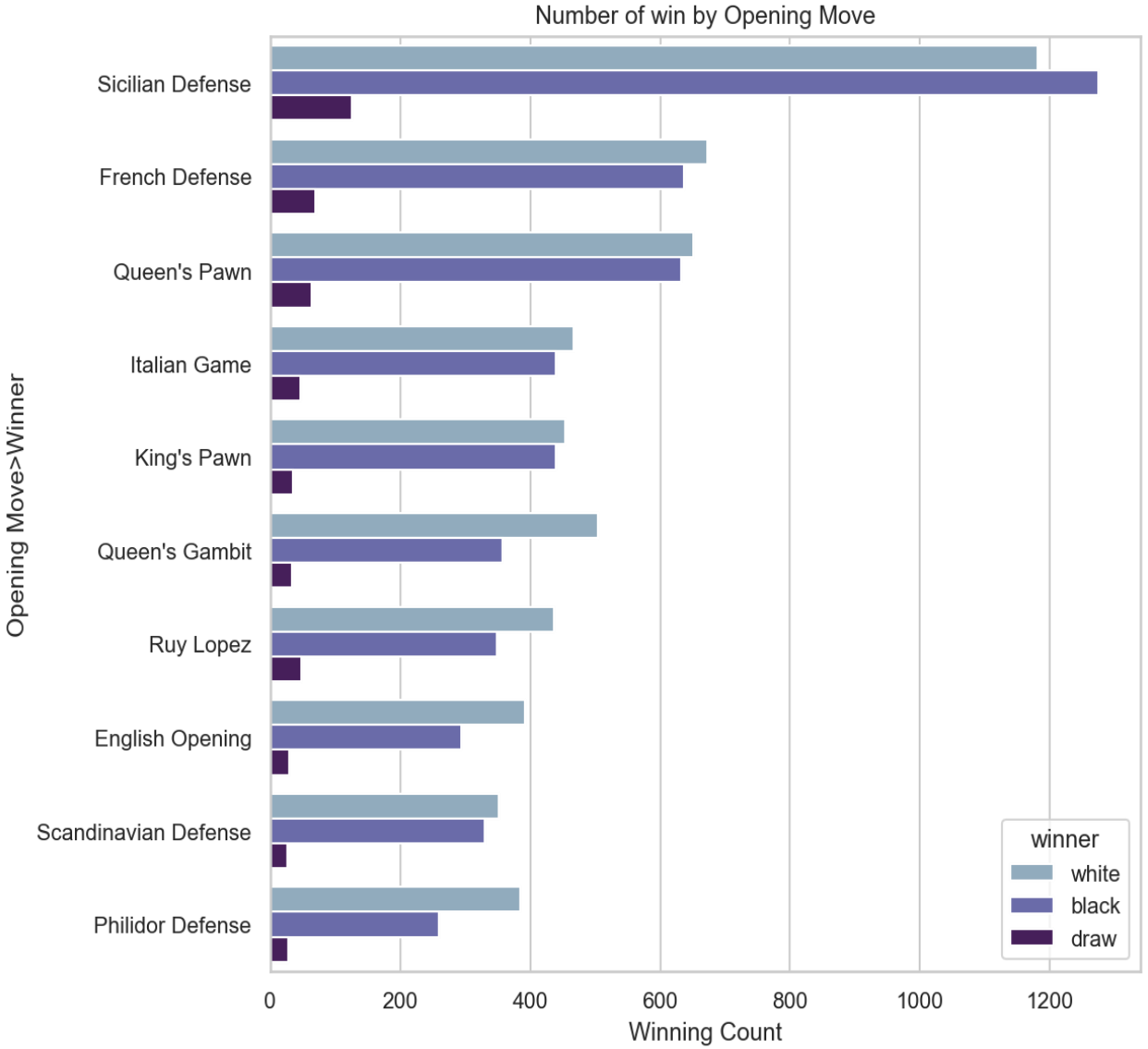
This a multi-classification problem



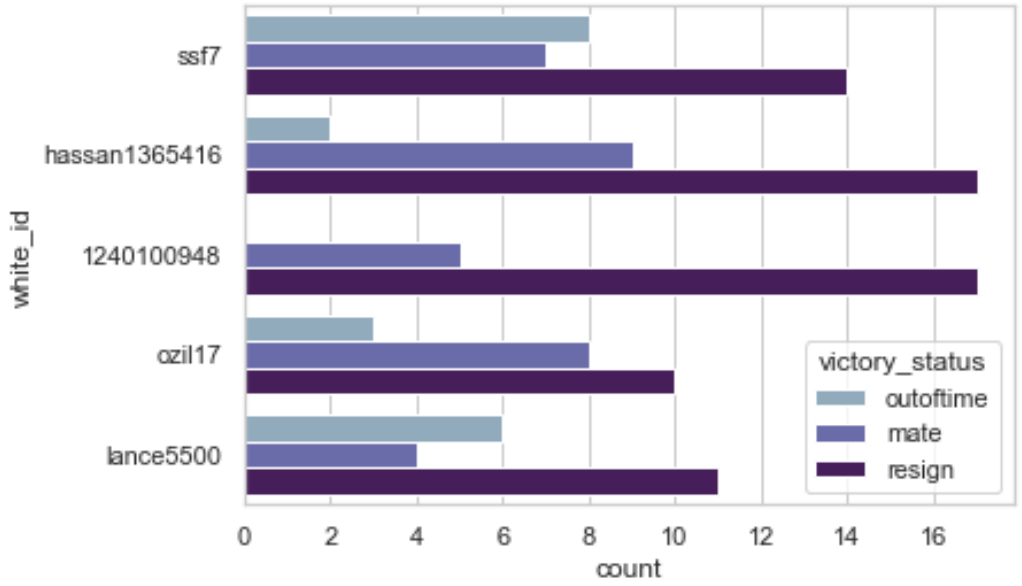
# Opening moves data

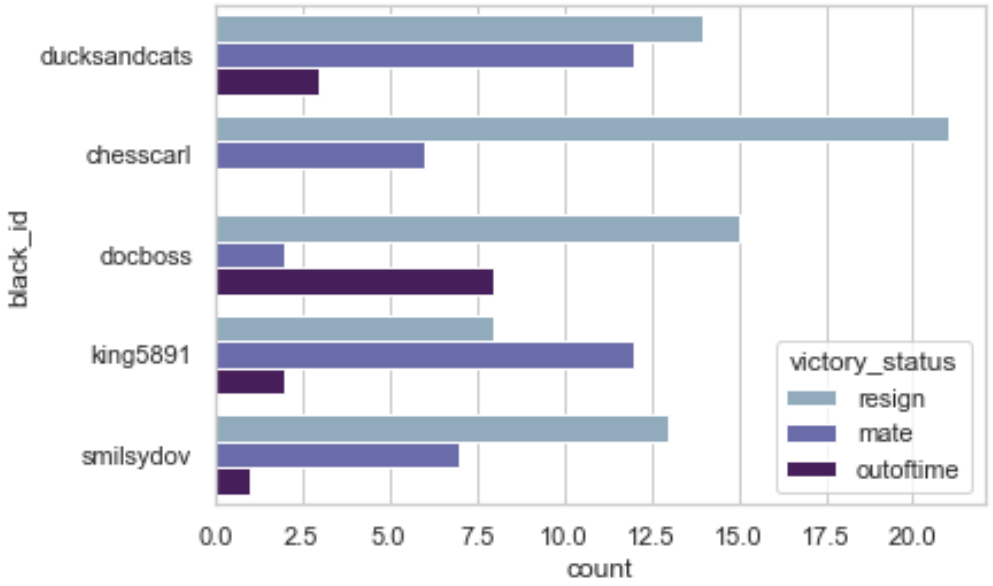



Sicilian Defense

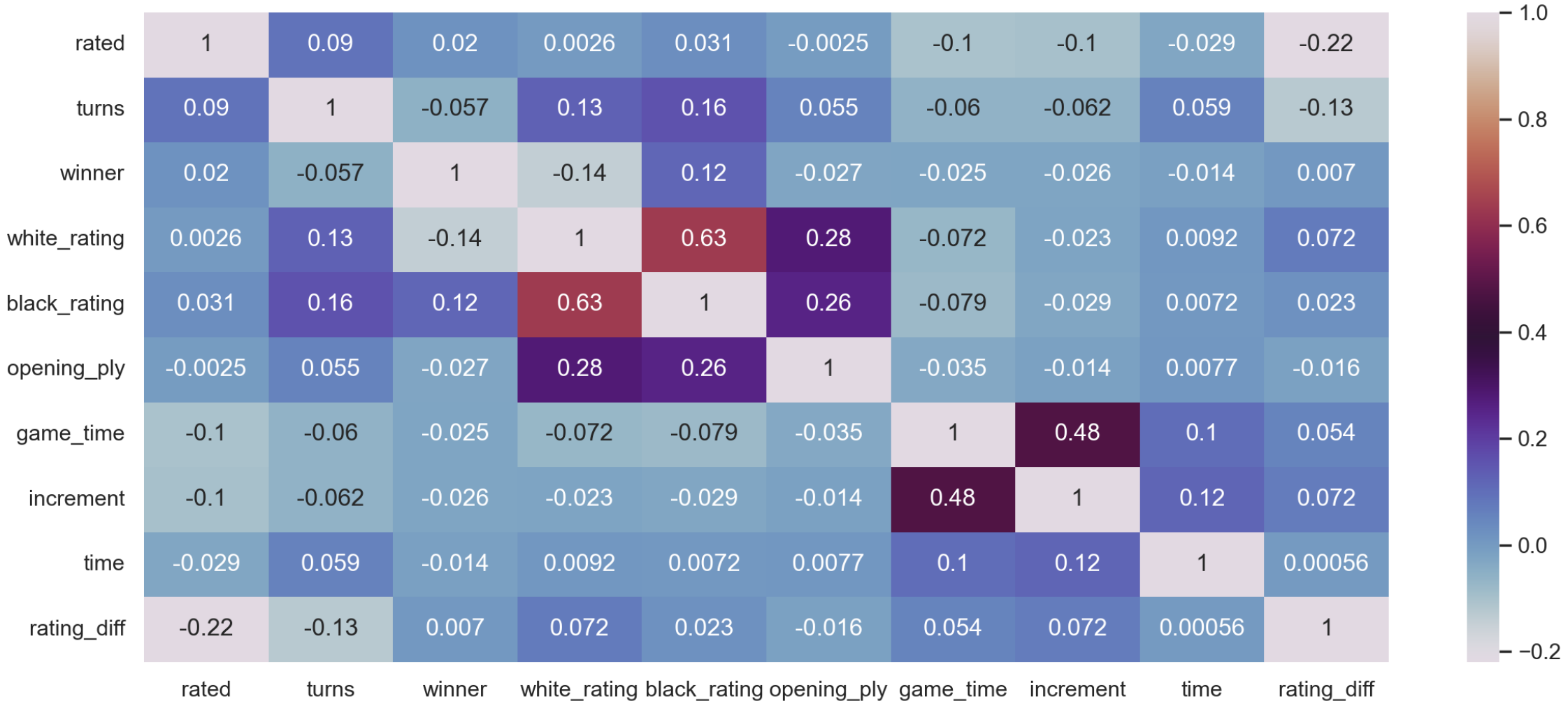


# Top White and Black Players



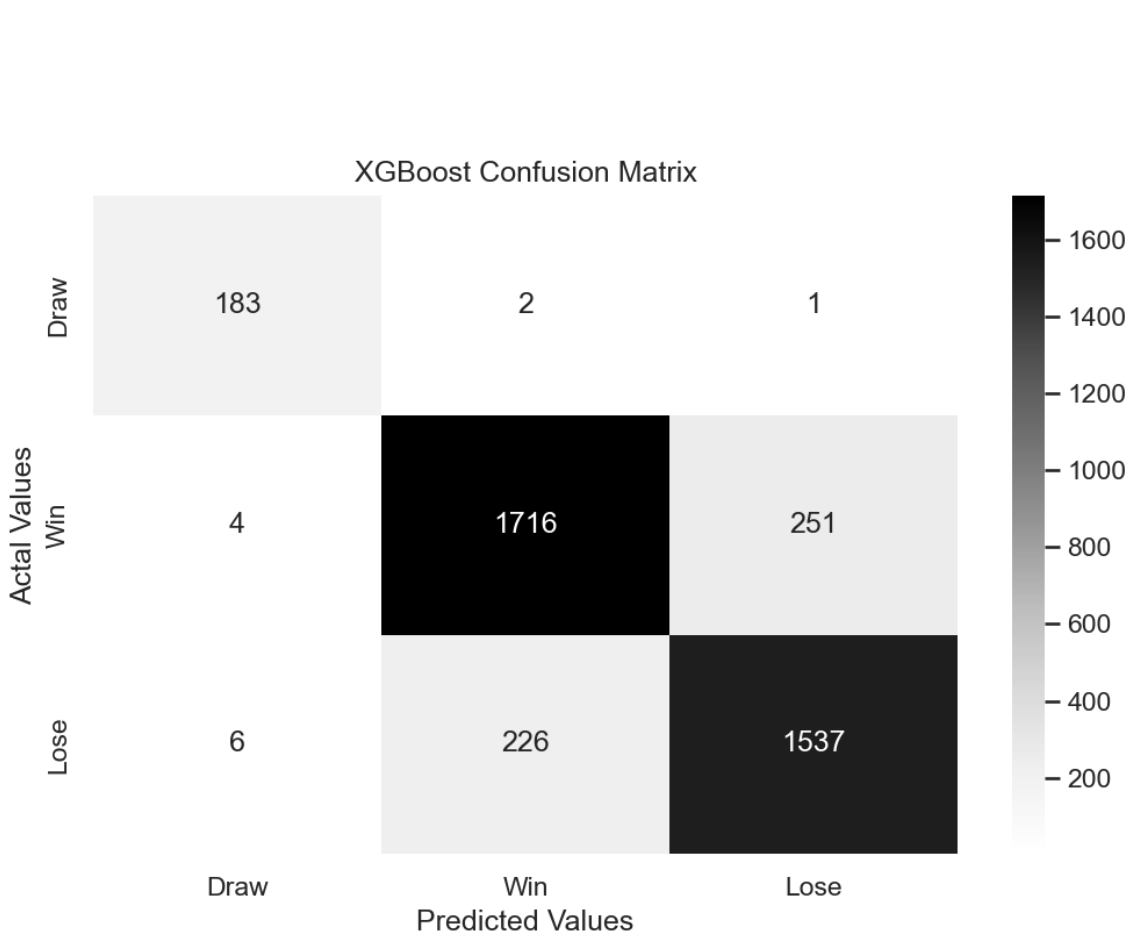
# Features Correlation Heatmap

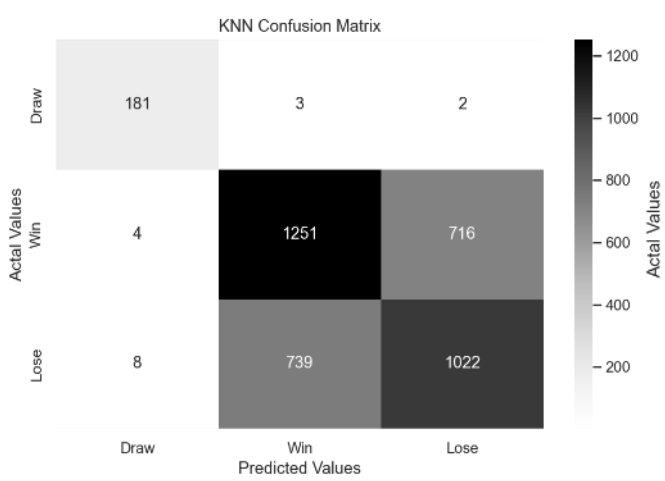


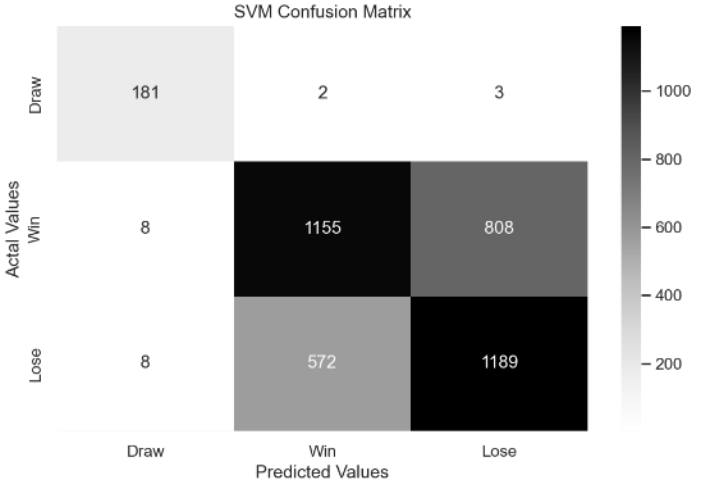
# Scorings

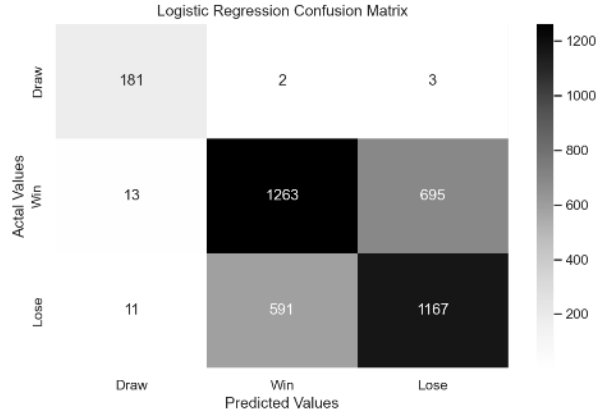
| MODELS | ACCURACY SCORE | F1 SCORE |
|---|---|---|
| Logistic Regression | 0.665053 | 0.664628 |
| Random Forest | 0.67244 | 0.672021 |
| XGBoost | 0.875191 | 0.875144 |
| SVM | 0.643148 | 0.642504 |
| KNN | 0.625064 | 0.624595 |

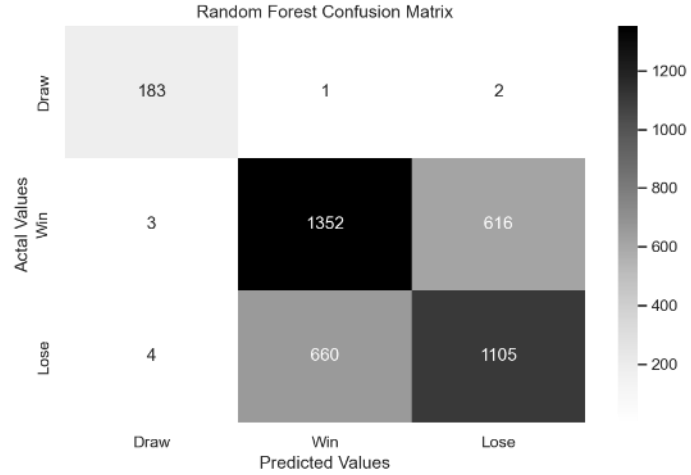
# Confusion Matrix



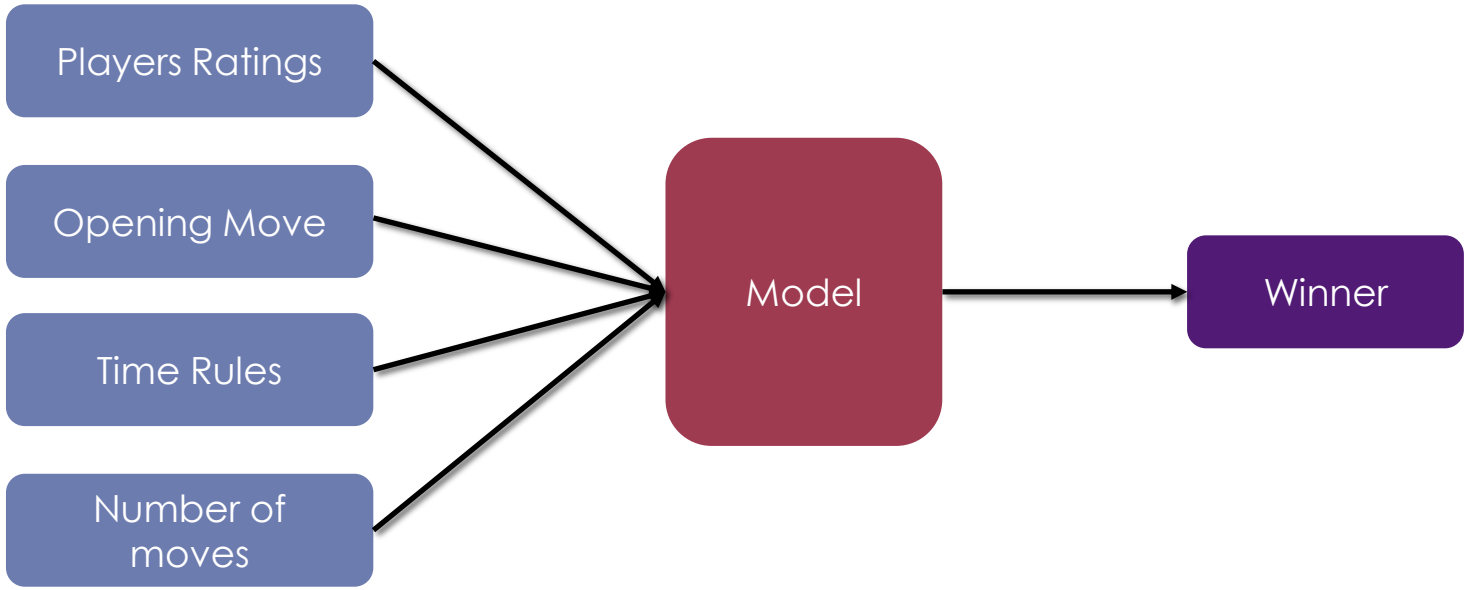



# Future Direction

- Connect the model with the original lichess.org API to be updated with realtime data.
- Deploy to production on a webapp using Flask.
- Apply the model on different sport games.
- Upgrade the model to including detailed movement analysis using on Neural Network models.



# Thank you

Any Questions?