## Problem Statement

You are given the data for evaluation of a partially finished chess game. You need to predict the winner of the game.

Create a model that predicts whether white wins denoted by (1) or black wins denoted by (0). For every game mentioned in this problem the winner went on to win the game normally (i.e. nobody won the game due to time-out). Each game was at least 20 moves long (i.e. both black and white played 20 moves each at least).

You are allowed to predict just the probability of white winning. So, an answer like 0.7 is acceptable for a game.

For predicting the result of the chess game, you are provided with the following information:

- Elo Rating of the both players
- Opening played in ECO code
- Evaluation of game at 10 moves ( where black has played the $10^{th}$ move)

Please refer this website to understand the eco code interpretation:

https://www.365chess.com/eco.php

## Data Format

The data format for both training and consumption by the final function is a csv with the following columns.

**The Training set** should be used to build your machine learning models. For the training set, we provide the target value (which of the two players went on to win the match) for each match. Your model will be based on the “features” as mentioned above. You can also use feature engineering to create new features

**The Test set** will be used to see how well your model performs on the unseen data. For the test set, we do not provide the target value. For each row in the test set, use your model to predict the target value (that is which of the two players won the match).

**Feature columns:**

- MatchID : The unique ID for each match
- WhiteElo : Elo ranking of white player
- BlackElo : Elo ranking of black player

- ECO: Opening played in ECO code. You can above link or Wikipedia to understand the interpretation of codes
- Eval: Evaluation of game at 10 moves (where black has played the 10th move).

**Target column:**

- Prob: Is the probability that white wins the match.

## Submission Format

You shall submit the following files:

- A word document describing your approach to the problem in a single page at the maximum as Q5_Explanation.docx
- A python file containing a function which consumes file paths for the data, prediction & model and saves your predictions to prediction file. A sample function is provided below as Q5.py.
- A pickled file containing all information you need from your trained model, we will run this trained model pickle file to generate predictions for out-of-sample data as Q5.pickle.
- The updated SampleSubmission.csv which was provided with this problem with your predictions in the format mentioned below as Q5.csv.It should contain 1500 rows in one column without header containing your prediction.

| MatchID | Prob |
|---|---|
| 10001 | 0.054576 |
| 10002 | 0.644805 |
| 10003 | 0.496124 |
| 10004 | 0.318032 |
| 10005 | 0.499279 |

| |
|---|
| 0 |
| 1 |
| 0 |
| 1 |
| 1 |

Put the files in a zip and upload the zip file. Do note that you need not put the files in a folder when zipping.

A sample python file with the format has been provided to you as Q5.py, you need to fill your logic in the function generate_model that saves a pickle file and uses pickle file to generate results.

## Evaluation Criteria

We shall measure submissions on the criteria:

Log-loss metric as the evaluation criteria.

Primarily we shall use the prediction file provided for test to form a cutoff.

We shall use extra out-of-sample data not provided to you for final scoring.