Project Report

On

EPL Ranking System

Course: INFO 6205

Team Members:

Aayush Jain (001401616)

<https://github.com/aayushjain92/INFO6205_Final_Project>



Contents

[Introduction 2](#_Toc38019828)

[What is Ranking System? 3](#_Toc38019829)

[Aim of the Project 3](#_Toc38019830)

[Datasets used in the Project 3](#_Toc38019831)

[Project Description 3](#_Toc38019832)

[Implementation 3](#_Toc38019833)

[Output 3](#_Toc38019834)

[Conclusion and Future Implementations 3](#_Toc38019835)

[References 4](#_Toc38019836)

# Introduction

## What is Ranking System?

Ranking system is an algorithm designed to evaluate the expression P(X**i**, X**j**) where X**i**, X**j** are elements from a set of competing elements X and P(X**i**, X**j**) is the probability that X**i** would beat X**j** if they met in a head to head matchup at neutral territory.

The application could be used to predict the rankings of the teams in a tournament and to get the likelihood of a team winning.

# Aim of the Project

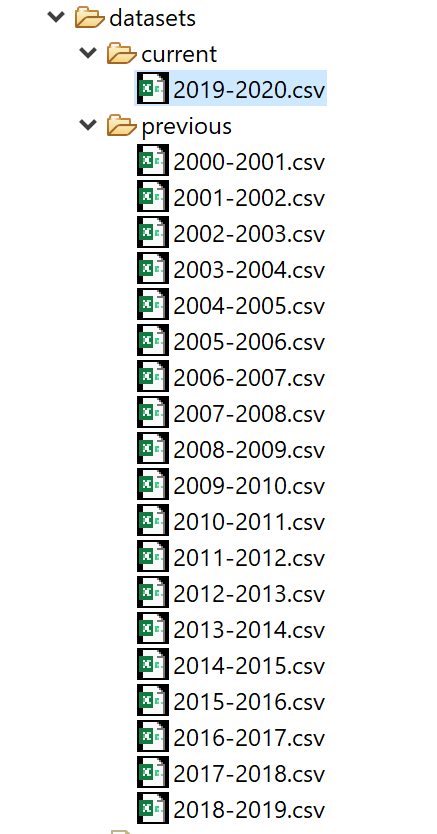
The 2019–20 Premier League is the 28th season of the Premier League, the top English professional [football](https://en.wikipedia.org/wiki/Association_football) league, since its establishment in 1992. The League was started on 9 August 2019 but since March, the season has been affected by the 2020 coronavirus pandemic and the remaining matches have been postponed.

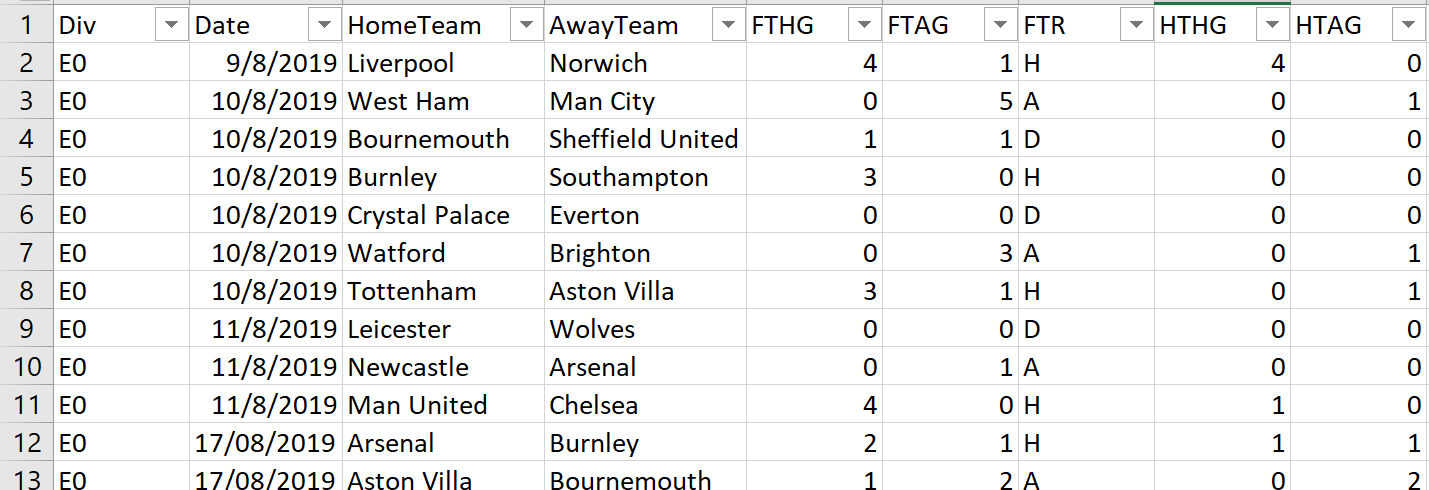
There are 20 clubs **in the Premier League**. During the course of a season (from August to May) each club **plays** the others **twice** (a **double** round-robin system), once at their home stadium and once at that of their opponents', for 38 games. Teams receive three points for a win and one point for a draw.

The Ranking System is an application designed to predict the outcome of the remaining matches that have been postponed and to predict the ranks of the clubs in this season.

# Datasets used in the Project

The data has been published on the following website. It contains the entire history of EPL (English Premiere League) since 2000.

<http://www.football-data.co.uk/englandm.php>

Fig: Sample screenshot of the data file

The historical data and the data of the current season has been separated into previous and current directories.

Following is the description of the columns used:

Div = League Division

Date = Match Date (dd/mm/yy)

Time = Time of match kick off

HomeTeam = Home Team

AwayTeam = Away Team

FTHG and HG = Full Time Home Team Goals

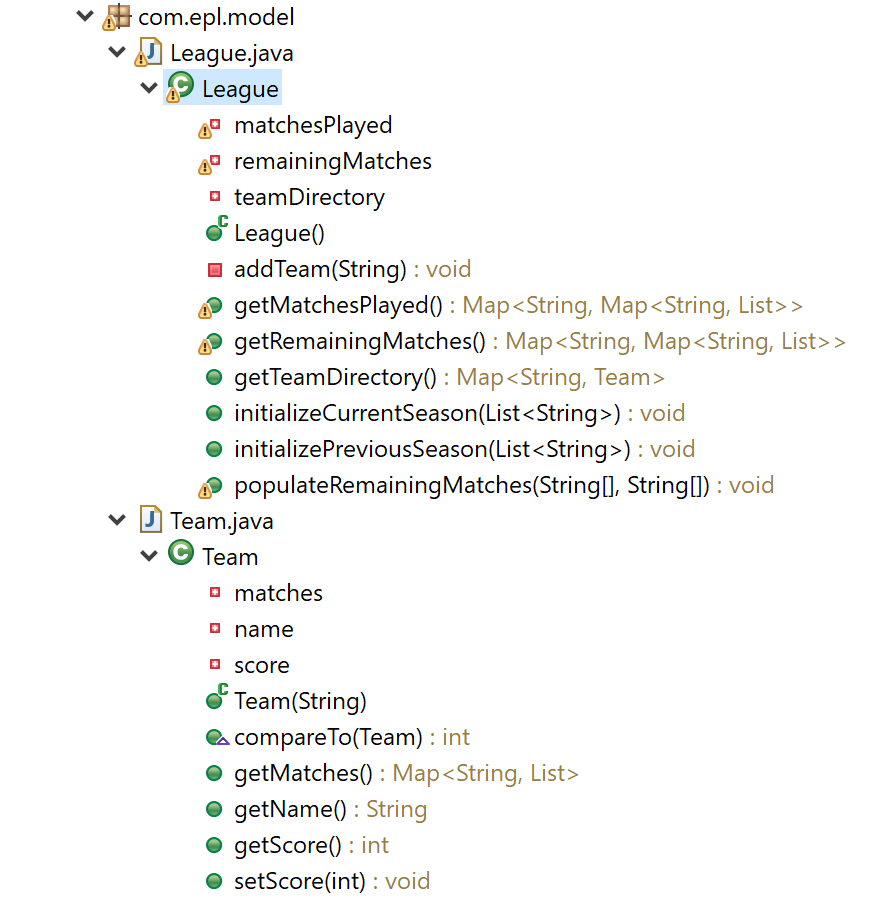
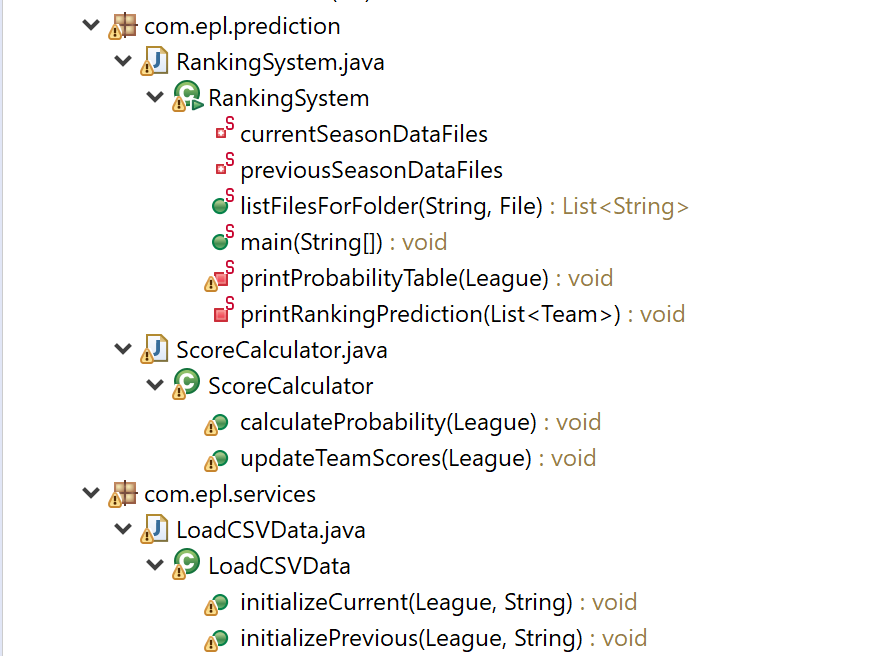
FTAG and AG = Full Time Away Team Goals

# Implementation

Let’s first get the overview of the code design:

Classes:

* com.epl.model
  + - League.java
      * The business logic is using League and Team Model to store and process the data
      * League contains three member variables i.e. a team directory which has 20 participating teams this season; matchesPlayed variable which contains all the info about the matches that have already been played; remainingMatches variable which contains all the info about the matches in the history because these matches have been postponed and are going to be predicted on the historical data
    - Team.java
      * It contains the team related information and the most important one is Score.
* com.epl.prediction
  + - Ranking System.java
      * It contains the main function
      * It also handles the printing functions to the console
    - ScoreCalculator.java
      * It contains the business logic and the merging of the Gaussian distributions to reach the most probable prediction
      * It uses the historical data to create a probability density function
* com.epl.services
  + - LoadCSVData.java
      * It loads the data from the CSV files in the model

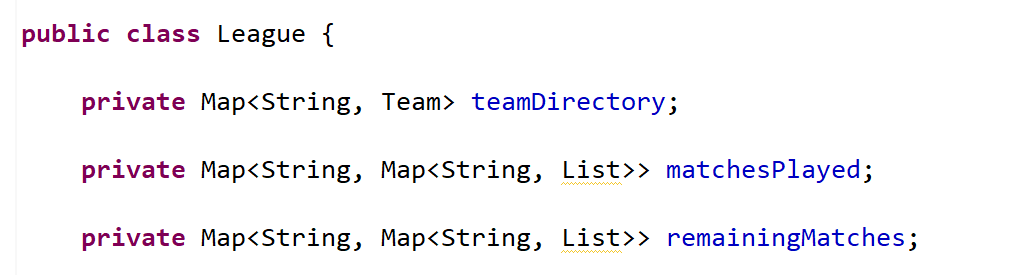


## Data Structures

Java.util.Map Collection has been heavily used to enable faster lookup of the data. The first “key” points to the home team and the next “key” points to the away team, making sure to reach the linked list of goal differences in O(1) time.

For Example:

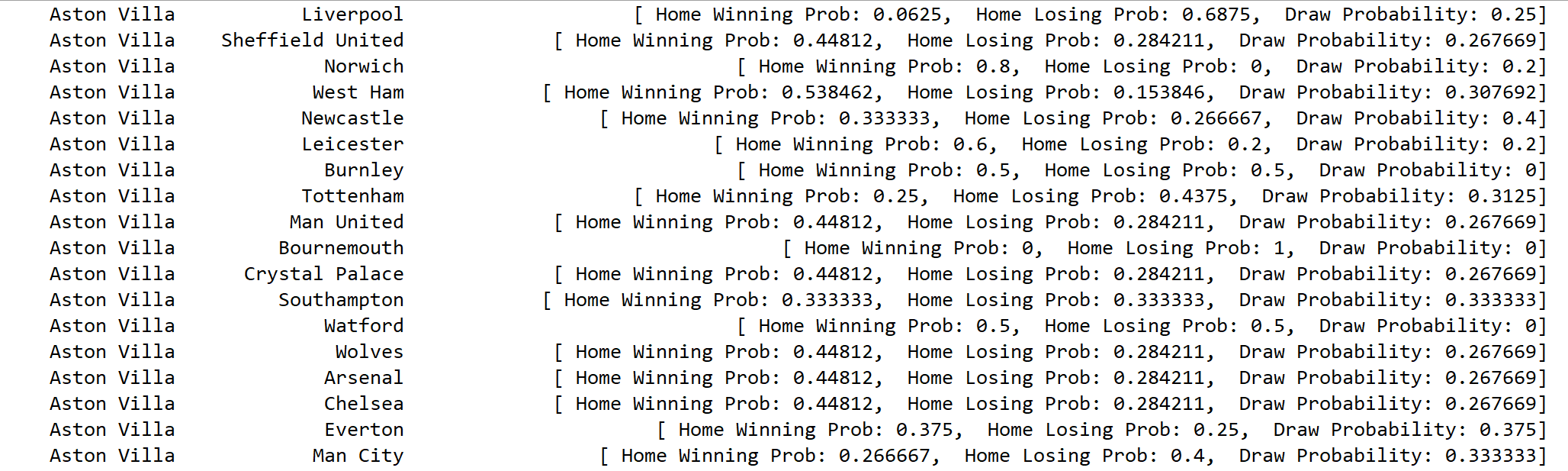
**remainingMatches.get(“Arsenal”).get(“Liverpool”)** will provide a list of goal differences between the two teams considering all the matches in the past since 2000.



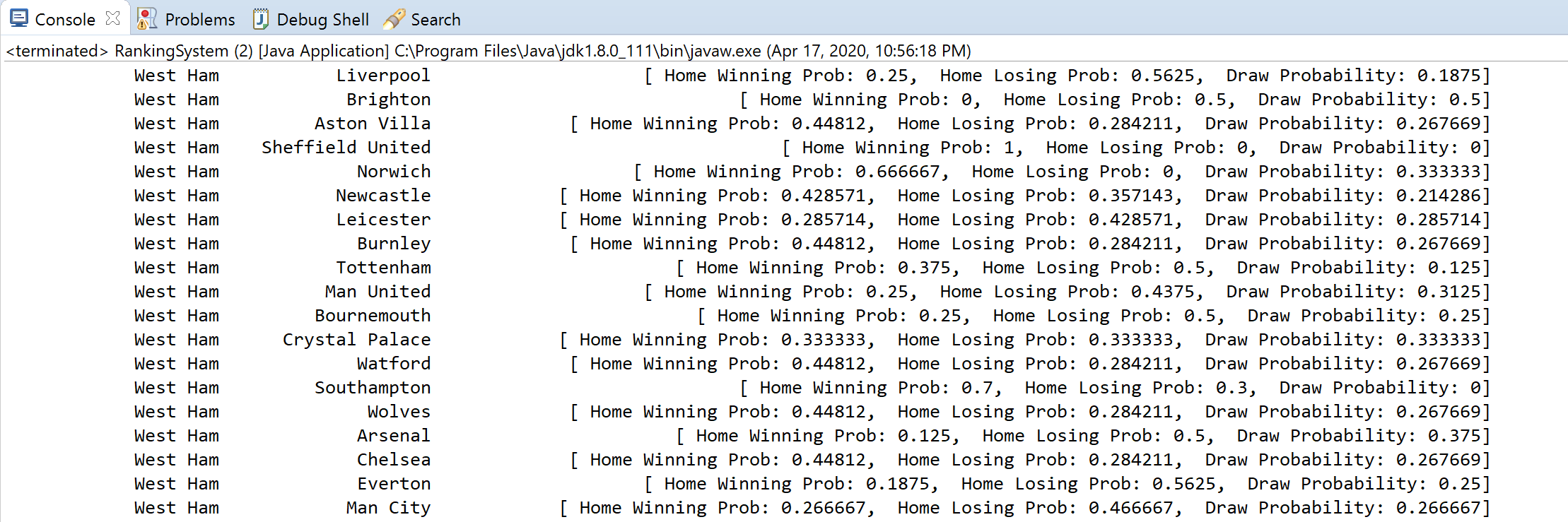
## Features:

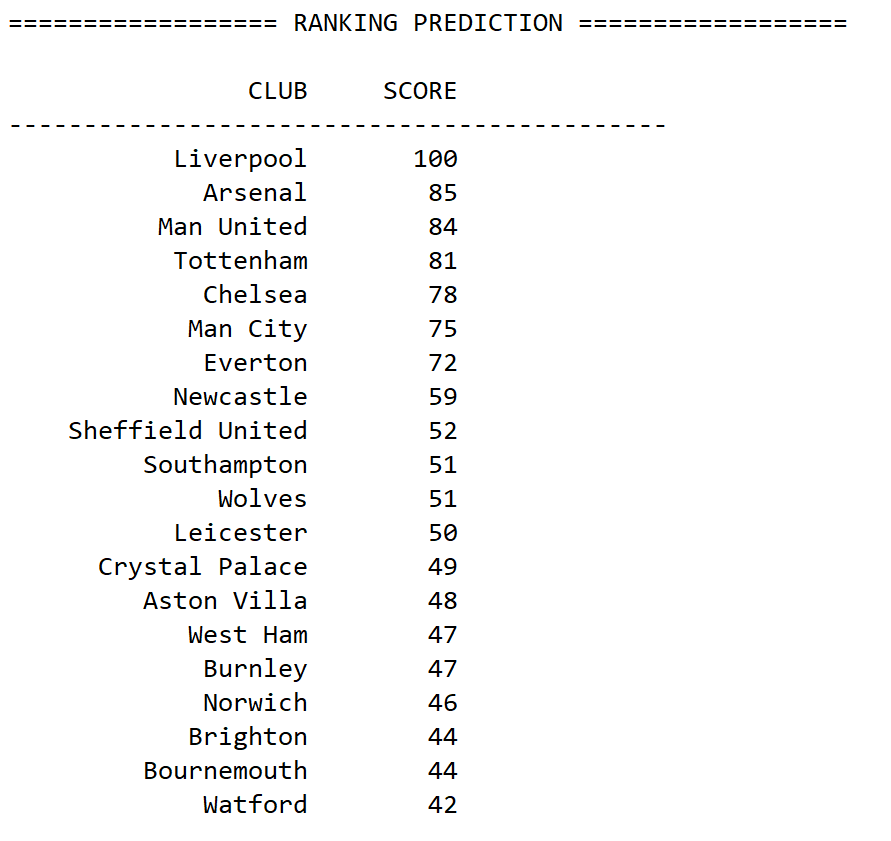
Considering home goals and away goals separately

# Output



The above output





# Conclusion and Future Implementations

# References