Premier League
Football Match Prediction

MUHAMMED SHIBIL C V
CB.SC.P2DSC23007

INTRODUCTION

Predicting the Winning Football Team

- Sports betting is a 500 billion dollar market (Sydney Herald)
- Football is played by 250 million players in over 200 countries (most popular sport globally).
- The English Premier League is the most popular domestic team in the world.
- Design a predictive model capable of accurately predicting if the home team will win a football match?

What is football match prediction?

Football match prediction is the process of forecasting the outcome of a football match before it takes place. This can be done using a variety of methods, including statistical analysis, machine learning, and expert judgment.

Why is football match prediction important?

Football match prediction is important for a variety of reasons. It can be used to:

- Inform betting decisions
- Guide team selection in fantasy football leagues
- Help teams develop strategies for upcoming matches
- Improve the accuracy of sports media coverage

OBJECTIVES

- Predict the outcome of football matches with a high degree of accuracy. This can be used to inform betting decisions, guide team selection in fantasy football leagues, help teams develop strategies for upcoming matches, and improve the accuracy of sports media coverage.
- Develop new and innovative methods for predicting football matches. This includes using new data sources, new machine learning algorithms, and new statistical techniques.
- Make football match prediction more accessible and user-friendly. This includes developing tools and resources that can be used by people of all levels of expertise.

LITERATURE REVIEW

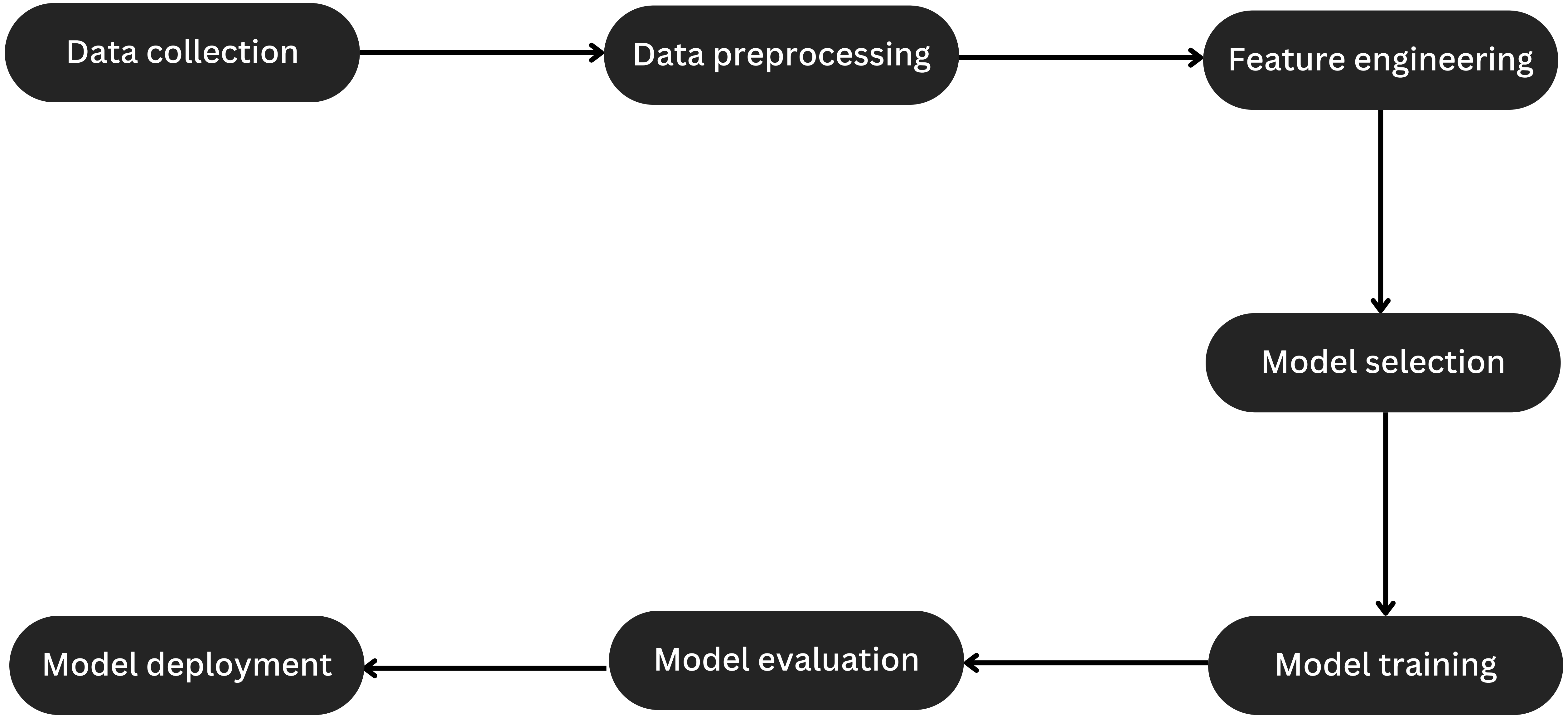
Football match prediction is a challenging task, but it has been the subject of much research in recent years. A variety of methods have been proposed for predicting the outcome of football matches, including statistical analysis, machine learning, and expert judgment.

Statistical analysis methods typically involve using historical data to develop models that can predict the outcome of future matches.

Some common statistical methods used for football match prediction include:

- Logistic regression: This method is used to predict binary outcomes, such as whether a team will win or lose.
- Decision trees: This method is used to create models that can predict the outcome of football matches by considering a variety of factors, such as team rankings and home/away advantage.
- Bayesian networks: This method is used to create models that can represent the complex relationships between the different factors that can influence the outcome of a football match.
- Support vector machines: This algorithm is used to create models that can classify data into different categories, such as "win" and "loss".

METHODOLOGY



Data collection

Data preprocessing

Feature engineering

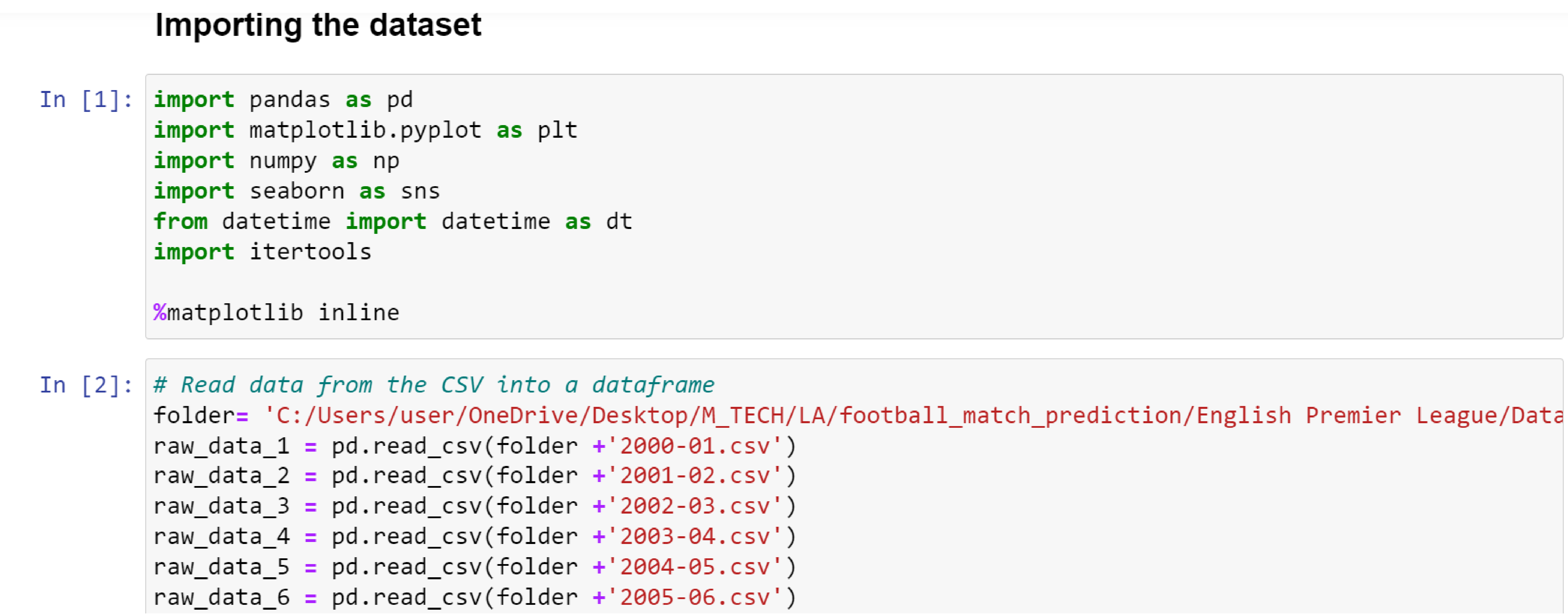
Model selection

Model training

Model evaluation

Model deployment

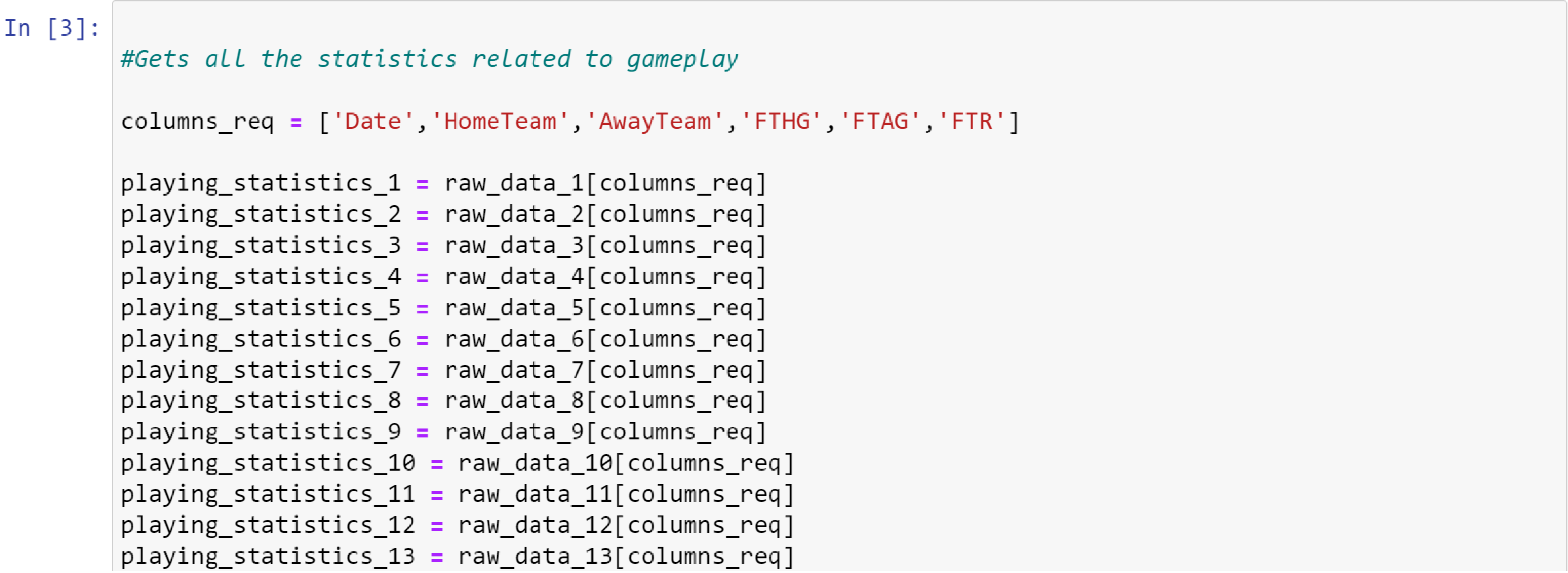
FOOTBALL MATCH PREDICTION



Importing the dataset

```
In [1]: import pandas as pd
        import matplotlib.pyplot as plt
        import numpy as np
        import seaborn as sns
        from datetime import datetime as dt
        import itertools

        %matplotlib inline
```

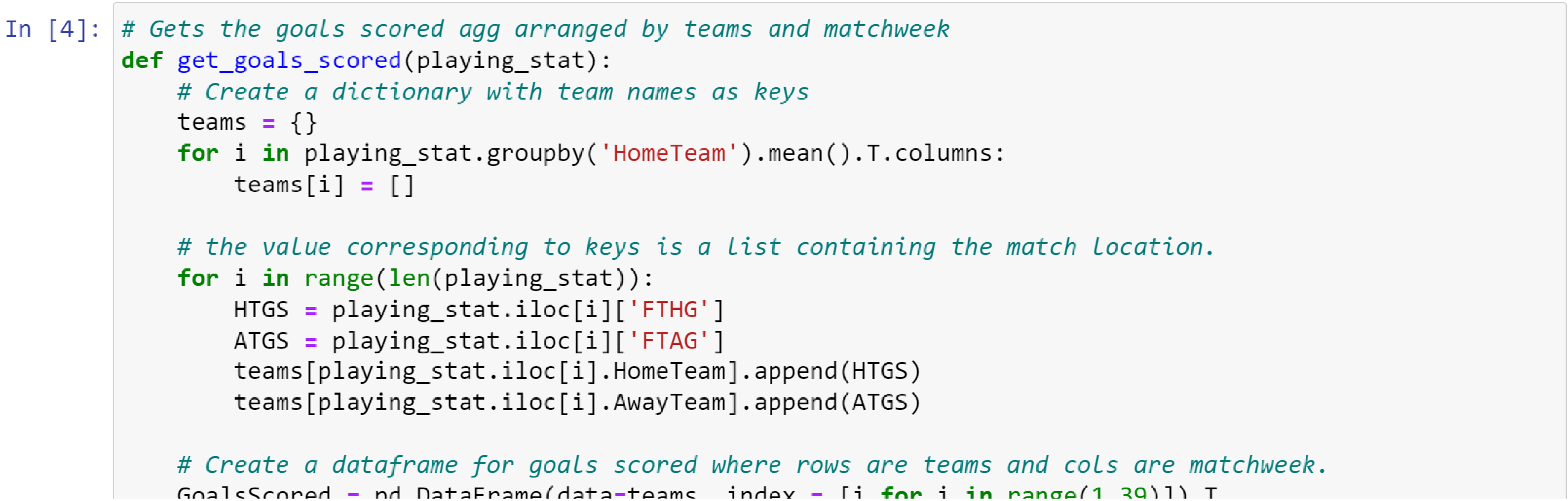
```
In [2]: # Read data from the CSV into a dataframe
        folder= 'C:/Users/user/OneDrive/Desktop/M_TECH/LA/football_match_prediction/English Premier League/Data
        raw_data_1 = pd.read_csv(folder +'2000-01.csv')
        raw_data_2 = pd.read_csv(folder +'2001-02.csv')
        raw_data_3 = pd.read_csv(folder +'2002-03.csv')
        raw_data_4 = pd.read_csv(folder +'2003-04.csv')
        raw_data_5 = pd.read_csv(folder +'2004-05.csv')
        raw_data_6 = pd.read_csv(folder +'2005-06.csv')
```



In [3]:

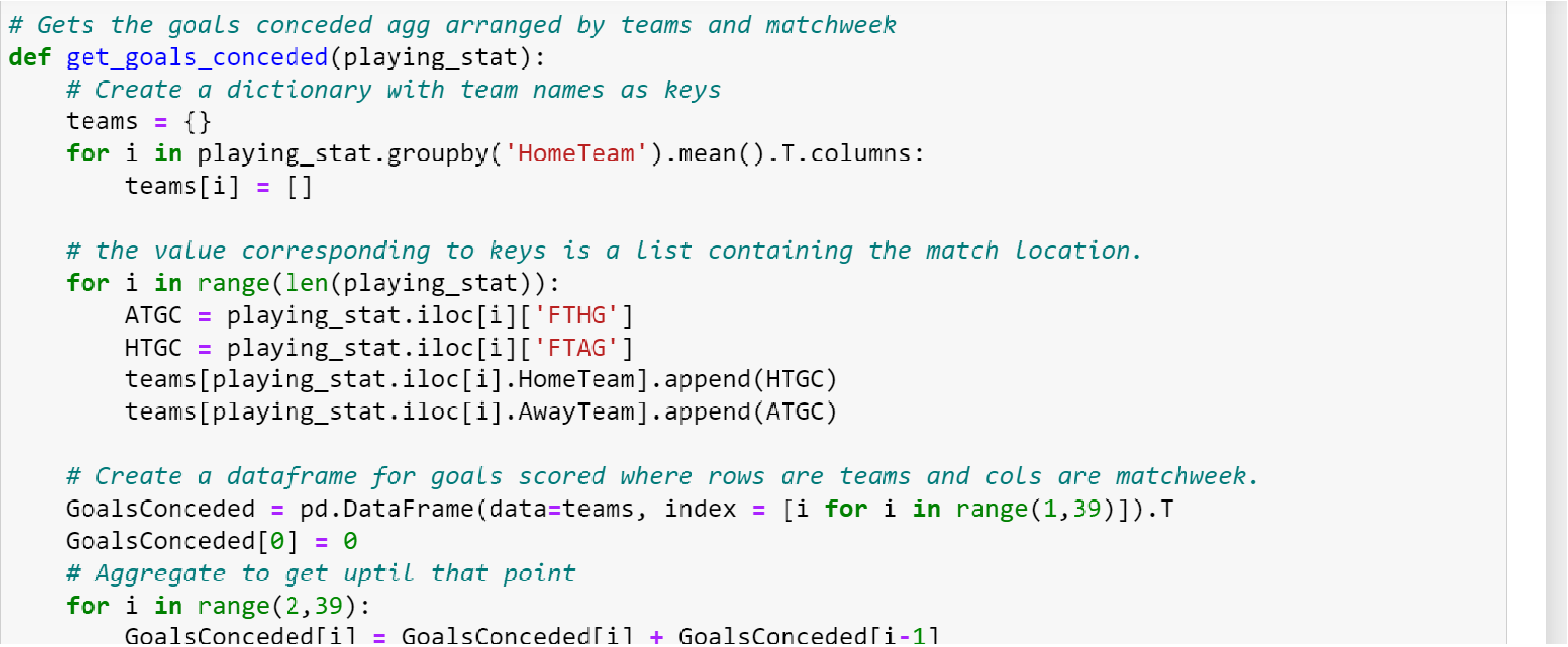
```
#Gets all the statistics related to gameplay

columns_req = ['Date','HomeTeam','AwayTeam','FTHG','FTAG','FTR']

playing_statistics_1 = raw_data_1[columns_req]
playing_statistics_2 = raw_data_2[columns_req]
playing_statistics_3 = raw_data_3[columns_req]
playing_statistics_4 = raw_data_4[columns_req]
playing_statistics_5 = raw_data_5[columns_req]
playing_statistics_6 = raw_data_6[columns_req]
playing_statistics_7 = raw_data_7[columns_req]
playing_statistics_8 = raw_data_8[columns_req]
playing_statistics_9 = raw_data_9[columns_req]
playing_statistics_10 = raw_data_10[columns_req]
playing_statistics_11 = raw_data_11[columns_req]
playing_statistics_12 = raw_data_12[columns_req]
playing_statistics_13 = raw_data_13[columns_req]
```

GOALS SCORED AND CONCEDED AT THE END OF MATCHWEEK, ARRANGED BY TEAMS AND MATCHWEEK

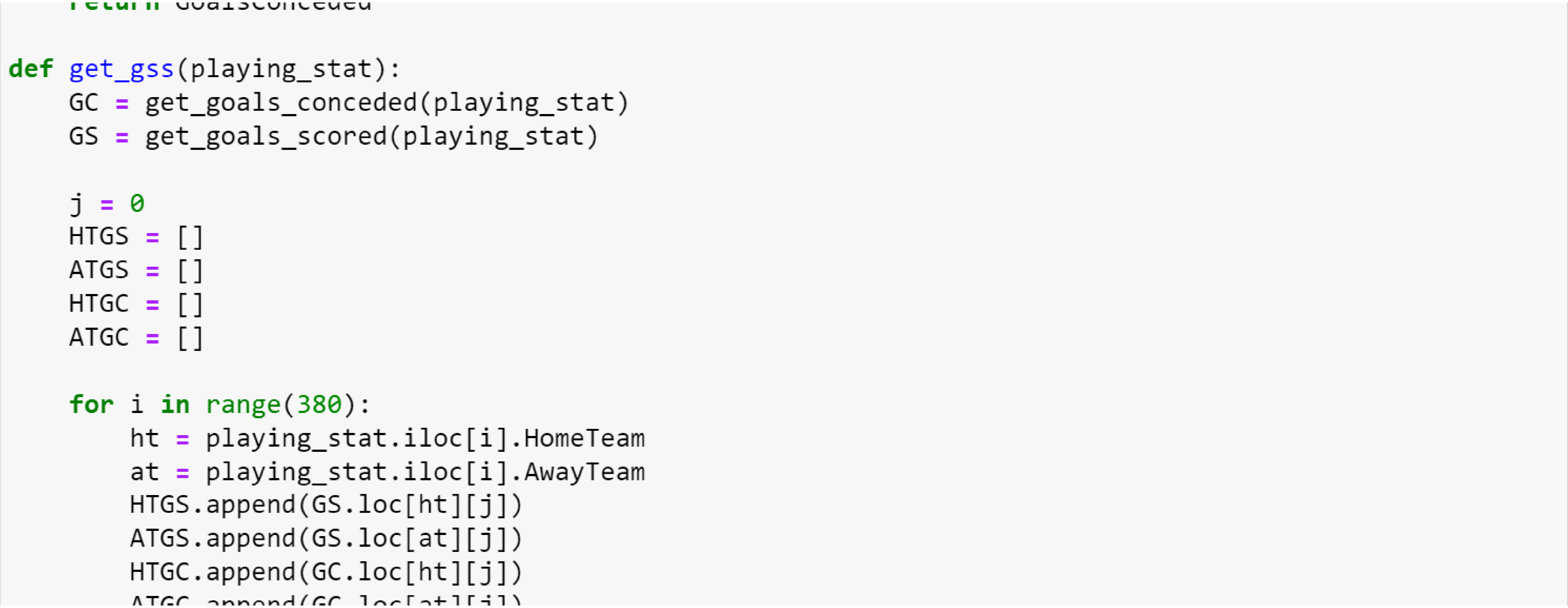


In [4]:

```
# Gets the goals scored agg arranged by teams and matchweek
def get_goals_scored(playing_stat):
    # Create a dictionary with team names as keys
    teams = {}
    for i in playing_stat.groupby('HomeTeam').mean().T.columns:
        teams[i] = []

    # the value corresponding to keys is a list containing the match location.
    for i in range(len(playing_stat)):
        HTGS = playing_stat.iloc[i]['FTHG']
        ATGS = playing_stat.iloc[i]['FTAG']
        teams[playing_stat.iloc[i].HomeTeam].append(HTGS)
        teams[playing_stat.iloc[i].AwayTeam].append(ATGS)

    # Create a dataframe for goals scored where rows are teams and cols are matchweek.
    GoalsScored = pd.DataFrame(data=teams, index = [i for i in range(1,39)]).T
```



```
# Gets the goals conceded agg arranged by teams and matchweek
def get_goals_conceded(playing_stat):
    # Create a dictionary with team names as keys
    teams = {}
    for i in playing_stat.groupby('HomeTeam').mean().T.columns:
        teams[i] = []

    # the value corresponding to keys is a list containing the match location.
    for i in range(len(playing_stat)):
        ATGC = playing_stat.iloc[i]['FTHG']
        HTGC = playing_stat.iloc[i]['FTAG']
        teams[playing_stat.iloc[i].HomeTeam].append(HTGC)
        teams[playing_stat.iloc[i].AwayTeam].append(ATGC)

    # Create a dataframe for goals scored where rows are teams and cols are matchweek.
    GoalsConceded = pd.DataFrame(data=teams, index = [i for i in range(1,39)]).T
    GoalsConceded[0] = 0
    # Aggregate to get uptil that point
    for i in range(2,39):
        GoalsConceded[i] = GoalsConceded[i] + GoalsConceded[i-1]
```



```
    return GoalsConceded

def get_gss(playing_stat):
    GC = get_goals_conceded(playing_stat)
    GS = get_goals_scored(playing_stat)

    j = 0
    HTGS = []
    ATGS = []
    HTGC = []
    ATGC = []

    for i in range(380):
        ht = playing_stat.iloc[i].HomeTeam
        at = playing_stat.iloc[i].AwayTeam
        HTGS.append(GS.loc[ht][j])
        ATGS.append(GS.loc[at][j])
        HTGC.append(GC.loc[ht][j])
        ATGC.append(GC.loc[at][j])
```

ENHANCEMENT

- Using more data
- Using more sophisticated machine learning algorithms
- Developing new features
- Ensembling models

RECENT ADVANCEMENTS AND FUTURE DIRECTIONS

- The use of real-time data: Football match prediction models are now able to use real-time data, such as player tracking data and social media sentiment data, to make more accurate predictions. This trend is likely to continue in the future, as more and more real-time data becomes available. For example, AI models could be used to predict the outcome of a football match while the match is still in progress.

REFERENCE

- J. Almulla and T. Alam, "Machine Learning Models Reveal Key Performance Metrics of Football Players to Win Matches in Qatar Stars League," in IEEE Access, vol. 8, pp. 213695-213705, 2020, doi: 10.1109/ACCESS.2020.3038601.
- A. Owen, "Dynamic Bayesian forecasting models of football match outcomes with estimation of the evolution variance parameter," in IMA Journal of Management Mathematics, vol. 22, no. 2, pp. 99-113, Apr. 2011, doi: 10.1093/imaman/dpq018.
- S. Kusmakar, S. Shelyag, Y. Zhu, D. Dwyer, P. Gastin and M. Angelova, "Machine Learning Enabled Team Performance Analysis in the Dynamical Environment of Soccer," in IEEE Access, vol. 8, pp. 90266-90279, 2020, doi: 10.1109/ACCESS.2020.2992025.

THANK YOU