Project document
ISTM 631: Programming for Business Analytics
Lomas (Ma Tao)

Program Purpose:

The program is used to simulate the result of a soccer game in England Premier League.

The simulation is based on the data of England Premier League season 2018-2019. The data stored in "2018-2019.csv" file. Each line is a certain game completed in real world. This file should be stored with the "soccer_simulation.py" in the same folder to be read correctly.

The program let the user to enter the name of the "Home team" and "Away team". Input validation is added to check whether the name of team is in the csv file.

Then it calculates the expected average number of goals of these two teams in a certain game. The program will utilize Poisson Distribution to simulate the number of goals of these two teams using expected number as 'lambda'.

The output will be the simulation result of the game, including the score of the game and who wins or draw game. The output will also print the real result stored in the csv file.

For example:

According to the csv file, Chelsea scored total 39 goals when they are home. Their expected average goals would be 2.05 per each home game. West Ham scored total 20 goals when they are away. Their expected average goals would be 1.05per each away game. Then program will simulate a random number using Poisson Distribution with lambda = 2.05 for how many Chelsea goals in this game (let's assume '3'), and do the same thing for West Ham (let's assume '1'). Finally, this program will compare the two-simulation results and shows "Chelsea 3 vs. 1 West Ham" and "Chelsea wins".

A game controller is also added inside the program to let user choose whether continue play this game. The user's input of 'Y' means continue; 'N' means end game.

Output sample:

```
This program is simulating a soccer game result.
Job start:
The teams' names are shown as follow:
Arsenal; Bournemouth; Brighton; Burnley; Cardiff; Chelsea; Crystal Palace; Everton;
Fulham; Huddersfield; Leicester; Liverpool; Man City; Man United; Newcastle;
Southampton; Tottenham; Watford; West Ham; Wolves;

Enter the name of home team: Chelsea
Enter the name of Away team: Arsenl
```

```
Can't find the team. Enter again: Arsenal
The simulation result is:
Chelsea 2 vs. 2 Arsenal
Draw game.
The real result of this game is:
Chelsea 3 vs. 2 Arsenal
Chelsea wins!

Do you wanna play again? ('Y' for Yes, 'N' for No): Y
Game continue...
Enter the name of home team: Chelsea
Enter the name of Away team: Liverpool
The simulation result is:
Chelsea 2 vs. 3 Liverpool
Liverpool wins!
The real result of this game is:
Chelsea 1 vs. 1 Liverpool
Draw game.

Do you wanna play again? ('Y' for Yes, 'N' for No): N
End Game.
Job Done...
```

Python Code:

```python
# python version 3.9.7
# ISTM 631 Programming for Business Analytics
# Lomas (Tao Ma)
# Nov 23, 2021
# Final Project
# soccer_simulation.py

'''
This program is used to simulate the game result of a soccer game.
It reads the game result of England Premier League from 2018-2019.csv.
It allows user to input the name of the two teams,
the Home team and Away team.
Then it calculates the expected number of goals of each team at Home and Away.
Then it simulate the goal by Poisson distribution using expected goals as lamda.
Then the simulate result of the game shows out.
It also shows out the result in real world for checking.
'''


'''
The File "2018-2019.csv" NEED be stored in the same folder as this py file!!!
'''


# import libraries
import csv # For reading csv file
```

```python
import numpy as np # For Poisson distrubution simulation

def main():
    '''This is Main function'''
    print('This program is simulating a soccer game result.')
    print('Job start:')
    '''Concepts in lecture 7: Create DataFrames From External Source'''
    # Call function 'getTeamName' and get the name of all teams
    Name_All = []
    Name_All = getTeamName('2018-2019.csv')
    print('The teams\' names are shown as follow:')
    '''Concepts in lecture 3: For Loop Statements'''
    for name in Name_All:
        print(name,end='; ')
    print('\n')
    Game_controller = True
    # The while loop for playing game again, unless input 'N' to end
    '''Concepts in lecture 3: While Loop Statements'''
    while Game_controller:
        # Let the user enter the name of Home and Away team
        HomeTeam = input('Enter the name of home team: ')
        while HomeTeam not in Name_All: # check validation:
            HomeTeam = input('Can\'t find the team. Enter again: ')
        AwayTeam = input('Enter the name of Away team: ')
        while AwayTeam not in Name_All: # check validation:
            AwayTeam = input('Can\'t find the team. Enter again: ')
        # Calculate the expected number of goals of Home team
        sum_Home_goals = 0
        sum_Away_goals = 0
        reader = csv.DictReader(open('2018-2019.csv','r')) # Read csv in a
dictionary
        Real_result = ''
        for line in reader:
            '''Concepts in lecture 5: Dictionaries'''
            if line['HomeTeam'] == HomeTeam:
                sum_Home_goals+=float(line['FTHG']) # FTHG stands for number of
goals Home team in a specific game
            if line['AwayTeam'] == AwayTeam:
                sum_Away_goals+=float(line['FTAG']) # FTAG stands for number of
goals Away team in a specific game
            if line['HomeTeam'] == HomeTeam and line['AwayTeam'] == AwayTeam: #
Check to match the real game result
                Real_result = HomeTeam+' '+line['FTHG']+' vs. '+line['FTAG']+'
'+AwayTeam # Real word result line
                Real_result +=
'\n'+game_result(HomeTeam,AwayTeam,line['FTHG'],line['FTAG'])
        Exp_avg_goal_H = sum_Home_goals/19 # Expected average goals per game of
Home team
        Exp_avg_goal_A = sum_Away_goals/19 # Expected average goals per game of
Home team
        # Call goals_simulation fuction to get simulate number of goals
```

```python
        HomeGoal = goals_simulation(Exp_avg_goal_H)
        AwayGoal = goals_simulation(Exp_avg_goal_A)
        # Output the simulation result
        print('The simulation result is:')
        print(f'{HomeTeam} {round(HomeGoal)} vs. {round(AwayGoal)} {AwayTeam}')
        print(game_result(HomeTeam,AwayTeam,HomeGoal,AwayGoal)) # call game_result
function to get result
        # Output the real game result
        print('The real result of this game is: \n'+Real_result)
        # Let the user to choose whether play this game again
        controller = input('\nDo you wanna play again? (\'Y\' for Yes, \'N\' for
No): ')
        Game_controller = play_again_control(controller)
    print('Job Done...') # End of the program


# Define functions
'''Concepts in lecture 5: Defining Simple Function'''
def getTeamName(csvFileName):
    '''Get each name of the team, and return a list'''
    csvFile = open(csvFileName,'r') # Open csv file
    csvReader = csv.reader(csvFile) # Read the data into a list
    '''Concepts in lecture 5: Lists'''
    TeamName = []
    for lines in csvReader:
        if csvReader.line_num == 1: # Ignore the 'Header' line
            continue
        TeamName.append(lines[2]) # Get team name of each line
    TeamName = list(set(TeamName)) # Delete duplicates team names using 'set'
    TeamName.sort() # Sort list by first letter
    csvFile.close()
    return TeamName

def goals_simulation(lamda):
    '''simulate the number of goals and return the result'''
    RND_Goal = np.random.poisson(lamda,1) # Simulation using poisson distribution
    return RND_Goal[0]

def game_result(Home,Away,HG,AG): # Home = HomeTeam, Away = AwayTeam, HG = Home
team goals, AG = Away team goals
    '''Check the result, return who wins/draw'''
    '''Concepts in lecture 3: Selection If and If-Else statements'''
    if HG == AG:
        return 'Draw game.'
    elif HG > AG:
        return Home+' wins!'
    elif HG < AG:
        return Away+' wins!'

def play_again_control(cmd):
    '''Controller of whether play again'''
    if cmd == 'Y': # 'Y' for continue game
```

```python
        print('Game continue...')
        return True
    elif cmd == 'N': # 'N' for end game
        print('End Game.')
        return False
    else: # Other input, terminate the program
        print('Wrong input! Program terminated.')
        return False

if __name__ == "__main__": main()
```