HW08 Baseball Statistics

Introduction

Once again, we will be using the Lahman Baseball Database in this week's assignment. We want to create a graph to visualize the relationship between how many RBIs a player has in a given year and their salary that same year. Unfortunately, Lahman provides the relevant data in two separate files. We will need to join the data to create the desired graph.

We have provided you with two CSV files named batting.csv and salaries.csv. The first, of course, contains the annual batting performance data from the last assignment. The second contains salary data for all Major League Baseball players dating back to the year 1985. You should download both files and place them in the same directory as your Python code for this assignment. *Caution!* Your program should work with any similarly formatted csv files.

Note: Before downloading batting.csv and salaries.csv, you should login https://c.zju.edu.cn/ firstly.

Reading in the data

First, you will need to read in the data files. In batting.csv, We are interested in *three* columns this week: `"playerID"`, `"yearID"` and `"RBI"`. Your program should skip the header in the file and completely ignore any lines where the RBI column does not contain a digit.

You should create an accumulator dictionary called `playeryear2rbis` that maps a tuple of the `"playerID"` string and `"yearID"` string to an integer representing the number of RBIs for that player and year. Just like last week, as you iterate through input the file, you should update the playeryear2rbis dictionary.

In salaries.csv, we are also interested in three columns: `"playerID"`, `"yearID"`, and `"salary"`. You should create a dictionary called `playeryear2salary` which maps a tuple of the `'playerID'` string and `'yearID'` string to an integer representing the player's salary for that year.

Preparing the data

After creating the two dictionaries, we need to join the data together and prepare it for plotting. We need to create two lists to hold the x and y values in our plot. We will call them `salaries` and `rbis`. You should iterate over all of the keys of `playeryear2salary` and use these keys to find the salary and rbi data for each player and each year. You should skip any `player,year` combinations that are not represented in the `playeryear2rbis` dictionary (in other words, the data for that player and year must be in both dictionaries.)

When you have the salary and RBI data for a given player and year, you should append them to the appropriate list. Once this loop is completed, you should end up with two lists of integers that have the exact same size. Each element in `salaries` will be an integer containing a player's salary for a certain year. The corresponding element in `rbis` (the one with the same index) will have the RBIs of that same player and year.

Plotting the data

Plot the data using the pyplot `plot` function. Make sure to `import matplotlib.pyplot as plt`. On the x axis, plot the `salaries`. On the y axis, plot the `rbis`. Use the format string `'k.'` (to plot the points as black dots). Title your plot `"Salary vs. RBIs in MLB"`. Label the X axis `"Salary"` and the Y axis `"RBIs"`. In your submission, you should **NOT** run `plt.show()`. You'd better print last 10 points like "[(x_{n-9}, y_{n-9}),(x_{n-8}, y_{n-8})…(x_n, y_n)]".

Submitting your solution

After you finish writing your code, you must submit it on http://others.zlcnup.com/cs101 before Dec 11th, 6pm. If you finish your code with Jupyter Notebook, you should create a hw08.py file, then paste your code into it. That's to say, you should upload your *.py file finally, other format is NOT allowed. You needn't rename your file, the system will check your information and rename your file. You'd better read the help information on website, we will firstly update on website if we have anything changed.

Note: During Wensday 8:00—18:00(LAB TIME), you can't upload your homework except lab's materials. The system will check it.

Using files and directories

In your final submission, you should use **open('batting.csv')** with no directory path in your code, on your own machine things may behave differently. Briefly, the *best* thing to do is to figure out where Python is running and move your file batting.csv there. Otherwise you can try to find out where your file is located and refer to it directly using the code shared at the end of lecture #13.