Final Report - ECE 3710 Project

Ammon Roberts - 10793687 Connor Paulson - 10684469

**Problem Solved/Explained:**

From the beginning of our project, the initial problem or question we wanted to solve was how to accurately predict the outcome of X amount of basketball games by using a small sample of data of statistical categories from a given basketball team and compare against the same data from the opponents they play next in their schedule. The small set of data we decided to use was Field Goal Percentage (FG%), 3-Point Percentage (3pt%), and Margin of Victory (MofV). Along with that, we also wanted to determine the Confidence Interval of each of these stats in an attempt to more accurately predict wins or losses based on a “modified” range of means in these sets of data.

We decided to use the local NBA team the Utah Jazz to gather stats about them and the teams they play against. To solve our problem of accurately predicting games, We used basketball-reference.com to grab data from the Utah Jazz’ first 5 games of the 2019-20 season. We stored the data of the aforementioned 3 statistical categories into a CSV file that we would later use in our Python script. We did the same for the next 5 opponents the Jazz play, and we grab their data from their first 5 games of the season. With these various CSV files containing the data of the first 5 games for each of these teams, we were able to find the means of all of the data and, as accurately as possible and with limited stat categories, predict the Utah Jazz’ record over their next 5 games, based on the stats of their first 5 compared with their opponent.

**Methods Used:**

We had a few main methods that we proposed that we would use and follow up on. These methods included Carefully analyzing data retrieved and input data into the Python script, Creating the Python script to analyze input data, and testing the Python script to verify accuracy and reliability of data and our processes.

We wanted to be careful with what data we used and how we used it, that is why we wanted to keep the data as similar as possible. That is why we took the same amount of games for each other team the Jazz play, mainly just for consistency of data among all the different teams being compared. We also had to be careful with how we passed the data into the various functions we call in the Python script. We created our Python script with the data in mind. We constructed the functions in our ***predict.py*** script to be consistent with what we were trying to do. We were also able to test it fairly simply, and did not have much trouble with the script running as planned, outside of random syntax errors that occasionally pop up.

**Analysis:**

We felt that the script that we designed runs well, and does what it’s overall intended function is supposed to do, which is from certain sets of data, predict the wins and losses the Jazz will have over their next 5 games. We had planned to take the sample mean of all the different sets of data for each team and then compare them to predict who is most likely to win. We had also planned to calculate the Confidence Interval of each data set. We realized that we would need extra data outside of what we had originally planned in order to test our data against the confidence interval for each game, so we shifted our focus to our main goal with keeping the data that we had originally and working with that. More detailed notes will be in observations.

During development, we noticed we had to keep modifying the way we were keeping track of data in order to ultimately compare at the end. We had to add other filters/lists to accurately keep record of the calculations we were making. Overall, it turned out to work well.

**Software Design:**

Our Script follows a functional approach to predicting wins and losses from the data that we get. Our Python script calls our main function load\_csv\_calculate\_and\_print that loads the CSV data, then will ultimately call a few other functions calculate\_mean, calculate\_confidence, compare\_stats,and get\_data to help calculate the sample mean and filter the data how we need in order to compare stats of previous games with other teams. We also have various global variables to help us keep track of wins, losses, and lists for each team. We also made it a point to make sure our script was well documented with comments describing all the necessary functions, as well as a comprehensive readme file.

**How to Run the Program:**

You must have Python 3.8 installed on your computer in order to run this script.

For instructions on how to accomplish this, visit https://www.python.org/.

For instructions installing pip, visit https://pip.pypa.io/en/stable/installing/.

To install the needed libraries, run "python -m pip install --user numpy scipy matplotlib" from the command line.

How to run the script:

1. Open a Command Window or terminal in the src folder

2. Run "python predict.py" or "python3 predict.py", depending on what version of Python your Python path variable points to.

This Python script automatically imports the raw data from the src/data folder.

The CSV files MUST be located in this folder, otherwise the script will not work.

The CSV files must also be saved without encoding options, as the script does not support encoding.

The script currently only supports a dataset of 5 items, but this can be changed if n is set equal to the number of items in a sample or dataset.

This script can also be further changed to work for data from other teams.

This can be done by doing the following:

1. Obtain the game data for a team of your choosing from https://www.basketball-reference.com/

2. Copy the data for field goals, three pointers, and margin of victory into a CSV, following the format of the included CSVs (including headers)

3. Edit the following lines of code (lines 145 - 149) and use the names of the CSVs you created:

load\_csv\_calculate\_and\_print("TEAM.csv")

load\_csv\_calculate\_and\_print("FIRSTOPPONENT.csv")

load\_csv\_calculate\_and\_print("SECONDOPPONENT.csv")

load\_csv\_calculate\_and\_print("THIRDOPPONENT.csv")

load\_csv\_calculate\_and\_print("FOURTHOPPONENT.csv")

load\_csv\_calculate\_and\_print("FIFTHOPPONENT.csv")

4. Create another CSV using the actual game data for the next 5 games, following the format of ACTUAL.CSV

You must track a win or loss using the characters 'W' or 'L', respectively

5. (NOT FULLY SUPPORTED) If you use the data for more than 5 games, you will need to update the following line of code (line 15) accordingly:

`

num\_games = n

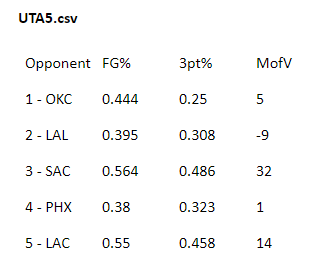
`

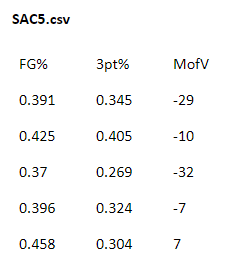
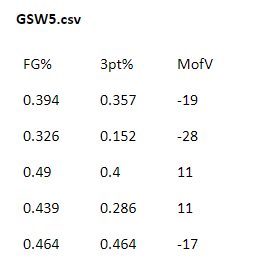
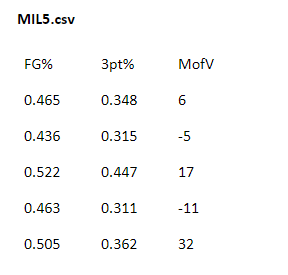
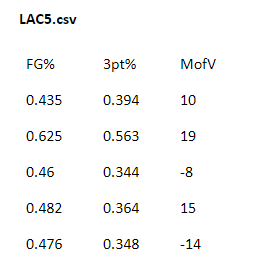
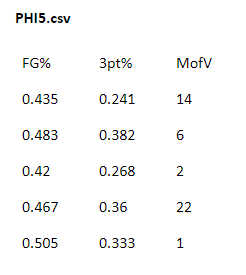
Where n is the number of games used in the sample

6. Run the script using the above instructions

END OF INSTRUCTIONS

**Figures/Plots/Data Tables:**

** **

** **

**Observations:**

As we developed our ideas as well as worked through our problems, we came to the conclusion that using the confidence interval wasn’t going to be very beneficial for what we were looking to do with our project. During the process of analyzing the data and observing the results of calculating the confidence interval, we realized that using a confidence interval on a small population of 5 was likely going to do more harm than good with regards to our prediction.

Because of this, we made the decision to use a simple mean to make our prediction. Using a simple mean, we were able to find the average number of games out of 5 that were won or lost, respectively. From there, we compared the resulting sets of data, and used conditional statements to decide if each game counted as a win or loss.

After allowing the script to run its course and to make the prediction, we found that based on the data we used, we were able to come up with a nearly accurate prediction. The script predicted that in the next 5 games, the Utah Jazz would win 4 games and lose 1 game. This was quite close to the actual count, being the Utah Jazz winning 3 games and losing 2 games.

Our observations showed that our prediction was somewhat accurate, but somewhat accurate isn’t quite accurate or even nearly accurate. We hypothesized that, given the time to gather more data and make some changes to our Python script, we would likely be able to use our script to make a more accurate prediction of how a team will perform.

The way to do this would be to allow for more games to be taken into account in our calculations. For example, instead of only using the data from 5 games, we could perhaps use data from 10 or 15 games, possibly even a whole season. If that were to be done, we would need to change the script somewhat in order to support a possibly dynamic number of games. We would also likely need to convert the calculations and output of the script to use a percentage, rather than just a win/loss count. We don’t expect that it would be particularly difficult, but it was not something we could do in the time we had allotted ourselves to complete the project.

**Conclusions:**

From the beginning of our proposal until now, we have been able to search out, find, create and use data from basketball statistics reflecting the Utah Jazz and some of the teams they have played. We were able to successfully define and use functions that helped us to calculate the sample mean, which we used to accurately predict how many games the Jazz will win or lose. We learned through experimentation and testing, that some parts of the project could not be handled with the size of the data that we had and the way we were testing it. We feel that we have successfully achieved our goal and have found a fairly reliable solution to our original problem as mentioned at the beginning, with the decided scope of the project.

**Output:**

**cpaulson@Cons-MBP src % python predict.py**

**UTA5.csv**

**~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~**

**Field Goal Mean:**

**Mean: 0.4666**

**Three Point Mean:**

**Mean: 0.365**

**Margin of Victory Mean:**

**Mean: 8.6**

**LAC5.csv**

**~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~**

**Field Goal Mean:**

**Mean: 0.48109999999999997**

**Three Point Mean:**

**Mean: 0.3838**

**Margin of Victory Mean:**

**Mean: 6.5**

**Jazz lose**

**MIL5.csv**

**~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~**

**Field Goal Mean:**

**Mean: 0.4782**

**Three Point Mean:**

**Mean: 0.3566**

**Margin of Victory Mean:**

**Mean: 7.8**

**Jazz win**

**SAC5.csv**

**~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~**

**Field Goal Mean:**

**Mean: 0.40800000000000003**

**Three Point Mean:**

**Mean: 0.3294**

**Margin of Victory Mean:**

**Mean: -14.2**

**Jazz win**

**GSW5.csv**

**~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~**

**Field Goal Mean:**

**Mean: 0.4226**

**Three Point Mean:**

**Mean: 0.3318**

**Margin of Victory Mean:**

**Mean: -8.4**

**Jazz win**

**PHI5.csv**

**~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~**

**Field Goal Mean:**

**Mean: 0.462**

**Three Point Mean:**

**Mean: 0.31679999999999997**

**Margin of Victory Mean:**

**Mean: 9.0**

**Jazz win**

**Jazz predicted record over their next 5 games (W/L): 4-1**

**Jazz actual record over their next 5 games (W/L): 3-2**