**CS240 Project Report**

**Name:** Mariem Badawy

**Student:** 215062020

**Introduction:**

In this report, I will be writing about my Python code I used to calculate statistics from the Teams.csv database which I used the Wins ‘W’ and Saves ‘SV’ columns from the year 1871 to the year 2016. I am going to analyze the data I used and the results I got and explain my code using the models from my results.

**Part 1:**

Brainstorm some questions:

* By using “Salaries.csv”, can we analyze in which league are players paid more? Is there any relationship between the leagues?
* By using “Teams.csv”, what is the relationship between the wins of the teams and their saves?

I chose the second question and decided to use the wins and saves columns from “Teams.csv” to compare them.

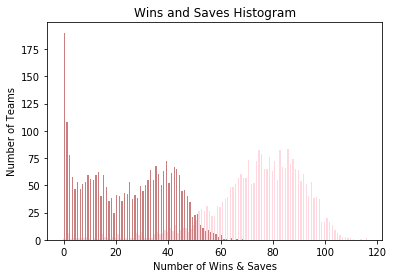
**Part 2:**

In this part of the project, I used pandas as pd to read the “Teams.csv” file, then I used the .dropna() built-in functions to choose the columns I will be using; Wins ‘W’ and Saves ‘SV’ columns. I then printed the columns.

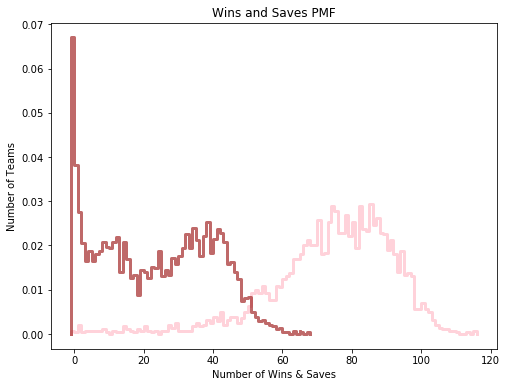
**Part 3:**

In this part, I calculated some relevant statistics to the columns and then I plotted a histogram, a PMF graph and a CDF graph.

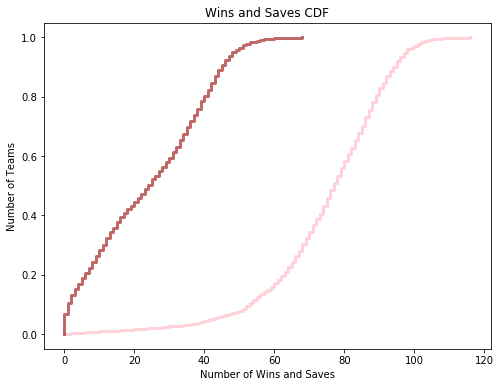
First, I calculated the minimum .min() and maximum .max() values in the wins column, the mean .mean() value of the wins column, and lastly the standard deviation .std() and variance .var() values of the wins column. Then, I calculated the exact same values for the saves column. I printed the results.



**Comment on Histogram:** This histogram shows how many teams had wins and saves at the same time and as you can see the more one of them increases the more the other decreases. The x axis is the number of wins and saves, and y axis is the number of teams who did both.



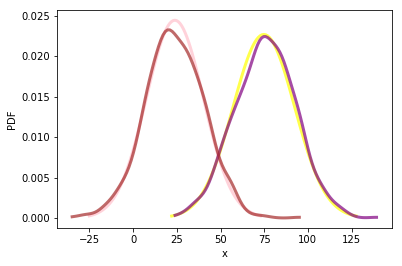
**Comment on PMF:** This PMF shows how many teams had wins and saves at the same time and as you can see the more one of them increases the more the other decreases. The x axis is the number of wins and saves, and y axis is the number of teams who did both.



**Comment on CDF:** This CDF shows how many teams had wins and saves at the same time and as you can see the more one of them increases the more the other decreases. The x axis is the number of wins and saves, and y axis is probabilities in of the teams in percentiles.

**Part 4:**

In this part of the project, I calculate the values of the means .mean(), standard deviations .std(), medians Median(), PDFs densities .Density() of the Wins and Saves and prints them in the end I used built-in functions to make the Normal PDF Distribution to see the density of the distribution. I plotted the Normal PDF Distribution. I also estimated (KDE) to find an appropriately smooth PDF that fits the data.



**Comment on PDF:** This is the Normal PDF Distribution results for the density of the distribution, the variables I used in the code were randomized by .random built-it function, I ran it various times and the results are too close.

**Part 5:**

In this part, I analyzed some questions about the relationship between the wins and the saves of the teams. If the team wins, does it mean also have saves? Is there a positive relationship or negative relationship? For answering this question, I used the covariance test which gives the tendency of two variables to vary together and correlation test that gives us the strength of the relationship between two relationships.

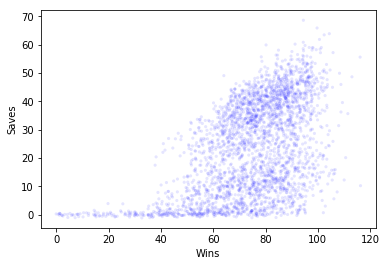
I used the covairance test Covariance () function and entered parameters wins and saves to get the covariance between the wins and the saves.

The result of the covariance test was 14528.15 which is a high number; this means that assets returns to move together.

This test wasn’t that useful so we have to try the correlation Correlation() function for conclusion.

The result of the correlation test was 50.59. Since *r* is positive, we say that the correlation is positive, which means that if one variable is high, the other tends to be high.

Then, I used the Jitter() function to plot the Jitter graph. I started with adding jitter which is a random noise added to the variable. Then I plotted by thinkplot.Scatter() and added the parameter heights and weights which have the wins and saves that are jittered by 0.5.



**Comment on Jitter Plot:** In the results of the Jitter Plot, we can see that the wins and saves have low similarity or at least not that high of a one. So when someone in a certain enters a save, the probability of the team losing is high.

**Part 6:**

Finally, in this part, I used the hypothesis test to the answer the question I chose. I got the function HypothesisTest() from the thinkstats2 module.

**Test Statistics:** There is a relationship between winning a game and entering a save. If the player in the team enters the save, the probability of the team winning is high.

**Null Hypothesis:** There is no relationship between winning a game and entering a save.

The result I got of the p-value was 0.0.

**Part 7:**

As a conclusion, I can say that there is a low similarity between entering a save and winning a game. When you observe the Histograms, PMfs, and CDFs, you find that the values are plotted on very different places. Later when we check the covariance and correlation we can conclude that the similarity is low. At the end when we ran the HypothesisTest, we saw that out test statistic’s results show that the wins and saves are inversly related. Finally, there is a negative relationship between wins and saves.

**Thank You**