Solution Design

1. **Create a csv file with a list of all presidents, their parties from 1920 onwards**
2. **Using Pandas load the .csv file into a Pandas dataframe**
3. **Download data from an appropriate financial website such as Google Finance, Yahoo Finance, Quandl, CityFALCON, or another similar source.**

I could not find historical data stretching 100 years for Dow Jones or SP500. I had to use data shared by another student on Piazza. I then created the file named **IndexFile**.csv in MS Excel, by aligning the presidents, with the years in power and their party

I then used pandas to load the file created in MS Excel named Index File into a data frame

ind = pd.read\_csv('IndexFile.csv')

1. **Calculate yearly returns for both the downloaded indices from 1920 onwards**

I used the following formula to calculate the returns

ind['SPCreturns'] = ind['SPC'] / ind['SPC'].shift(1)

ind['DJIreturns'] = ind['DJI'] / ind['DJI'].shift(1)

1. **Segregate returns in terms of Presidency – i.e. stock market returns during Democratic and Republican years**

I then used groupby function to segregate the returns for both party and president

indgroupParty = ind[['President', 'SPCreturns','DJIreturns']].groupby('President').sum()

indgroupParty = ind[['Party', 'SPCreturns', 'DJIreturns']].groupby('Party').sum()

1. Calculate measures of central tendency (mean return, median return, variance of returns) for each of the two groups.

indgroupParty = ind[['Party', 'SPCreturns', 'DJIreturns']].groupby('Party')

print(indgroupParty.describe())

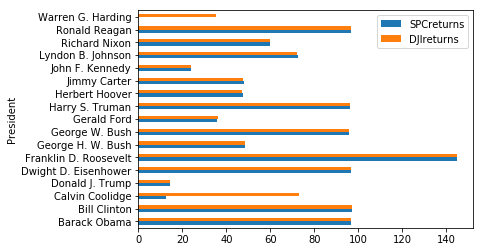
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **DJIreturns** | |  |  |  | \ |  |  |
|  | count | mean | std | min | 25% | 50% | 75% | max |
| **Party** |  |  |  |  |  |  |  |  |
| **Democratic** | 576 | 1.005242 | 0.048672 | 0.763265 | 0.980045 | 1.008177 | 1.32703 | 1.401808 |
| **Republican** | 601 | 1.002614 | 0.056414 | 0.69762 | 0.977168 | 1.004788 | 1.33092 | 1.358287 |
|  |  |  |  |  |  |  |  |  |
|  | **SPCreturns** | |  | \ |  |  |  |  |
|  | count | mean | std | min | 25% | 50% | 75% | max |
| **Party** |  |  |  |  |  |  |  |  |
| **Democratic** | 576 | 1.005768 | 0.051582 | 0.749533 | 0.977979 | 1.008764 | 1.033992 | 1.422156 |
| **Republican** | 506 | 1.001283 | 0.057271 | 0.705232 | 0.97534 | 1.003317 | 1.031407 | 1.385638 |

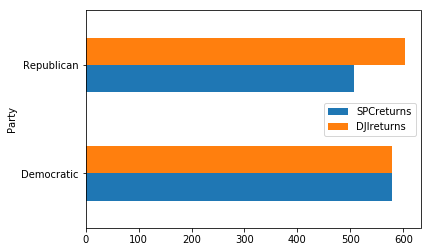
1. **Represent the findings through suitable comparative graphical studies**

I used Matplotlib plot to plot the graphs

indgroupParty.plot(kind='barh')

plt.show()





**Technical Aspects:**

The project was carried out on a Window 10 Pro computer using Anaconda 3, Spyder 3.2.6 IDLE and Python 3.6. The libraries used were from Pandas imported as pd and Matplotlib.pyplot imported as plt to graphically show the graphs.