**Assignment 1- Part 2**

**PCA Assignment**

1. **Problems and Solution**
   1. Load the data into a pandas dataframe.

Q1. Load the data

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

date open high low close volume Name

0 08/02/2013 15.07 15.12 14.63 14.75 8407500 AAL

1 11/02/2013 14.89 15.01 14.26 14.46 8882000 AAL

2 12/02/2013 14.45 14.51 14.10 14.27 8126000 AAL

3 13/02/2013 14.30 14.94 14.25 14.66 10259500 AAL

4 14/02/2013 14.94 14.96 13.16 13.99 31879900 AAL

... ... ... ... ... ... ... ...

619035 01/02/2018 76.84 78.27 76.69 77.82 2982259 ZTS

619036 02/02/2018 77.53 78.12 76.73 76.78 2595187 ZTS

619037 05/02/2018 76.64 76.92 73.18 73.83 2962031 ZTS

619038 06/02/2018 72.74 74.56 72.13 73.27 4924323 ZTS

619039 07/02/2018 72.70 75.00 72.69 73.86 4534912 ZTS

[619040 rows x 7 columns]

* 1. Sort names in alphabetical order. How many are there? List the first and last 5 names.

Q2. Sorting names

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Total number of unique names 505

first 5 names

A AAL AAP AAPL ABBV

last 5 names

ZTS ZION ZBH YUM XYL

* 1. Remove names whose first date is after 1st Jan 2014 or the last date is before 31st Dec 2017.
     1. Which names were removed?
     2. How many are left?

Q3. Names with first date after 1/1/2014 or last date before 31/12/2017

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

APTV , BHF , BHGE , CFG , CSRA , DWDP , DXC , EVHC , FTV , GOOG , HLT , HPE , HPQ , INFO , KHC , NAVI , PYPL , QRVO , SYF , UA , WLTW , WRK ,

Number of elements removed 22

Number of elements remaining 483

* 1. Identify the dates that are common and remove which are before 1st Jan 2014 or after 31st Dec 2017.
     1. How many dates are there?
     2. What are the first and last 5 dates?

Q4. Dates that are common to all the remaining names

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Total number of dates 607342

Number of dates removed that are before 1/1/2014 or after 31/12/2017 1050

Number of dates in the selected period range 994

first 5 dates in the range

02/01/2014

03/01/2014

06/01/2014

07/01/2014

08/01/2014

last 5 dates in the range

29/12/2017

28/12/2017

27/12/2017

26/12/2017

22/12/2017

* 1. Build a new pandas dataframe which has a column for the names and a row for the dates
     1. Dataframe populatedwith the “close” values for each corresponding name and date.

Q5. Dataframe created with dates in row and names in column

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

AAL AAPL AAP ABBV ABC ABT ACN ... XRAY XRX XYL YUM ZBH ZION ZTS

Date ...

02/01/2014 25.360 79.0185 109.74 51.98 69.89 38.23 81.13 ... 65.90 54.79 58.91 19.08 132.00 102.64 34.00

03/01/2014 26.540 77.2828 112.88 52.30 69.94 38.64 81.40 ... 66.11 54.90 58.70 19.09 134.14 103.06 34.43

06/01/2014 27.030 77.7042 111.80 50.39 69.69 39.15 80.54 ... 64.12 55.31 58.62 19.16 138.53 103.07 34.13

07/01/2014 26.905 77.1481 113.18 50.49 70.45 38.85 81.52 ... 64.66 55.50 58.83 19.48 138.98 104.42 34.12

08/01/2014 27.630 77.6371 112.30 50.36 71.14 39.20 82.15 ... 65.63 56.18 58.15 19.48 132.00 103.61 33.92

... ... ... ... ... ... ... ... ... ... ... ... ... ... ... ...

22/12/2017 52.590 175.0100 100.55 98.21 92.46 56.93 153.89 ... 52.70 59.69 18.94 134.98 102.03 32.72 92.98

26/12/2017 52.850 170.5700 101.96 97.75 93.25 57.00 152.99 ... 53.28 59.65 19.02 130.42 102.10 33.15 93.82

27/12/2017 52.400 170.6000 99.77 98.09 92.60 57.47 153.32 ... 53.52 59.73 19.00 129.85 102.73 33.40 94.13

28/12/2017 52.460 171.0800 99.71 97.79 92.59 57.46 153.57 ... 52.96 59.12 18.96 133.55 102.95 33.08 93.89

29/12/2017 52.030 169.2300 99.69 96.71 91.82 57.07 153.09 ... 53.98 59.50 19.18 133.17 102.49 33.82 96.73

[994 rows x 483 columns]

* 1. Create another dataframe containing returns calculated

Q6. Dataframe created with returns

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

AAL AAPL AAP ABBV ABC ... XYL YUM ZBH ZION ZTS

Date ...

03/01/2014 0.046530 -0.021966 0.028613 0.006156 0.000715 ... -0.003565 0.000524 0.016212 0.004092 0.012647

06/01/2014 0.018463 0.005453 -0.009568 -0.036520 -0.003574 ... -0.001363 0.003667 0.032727 0.000097 -0.008713

07/01/2014 -0.004624 -0.007157 0.012343 0.001985 0.010905 ... 0.003582 0.016701 0.003248 0.013098 -0.000293

08/01/2014 0.026947 0.006338 -0.007775 -0.002575 0.009794 ... -0.011559 0.000000 -0.050223 -0.007757 -0.005862

09/01/2014 0.064785 -0.012772 0.011131 0.017077 0.003374 ... 0.009802 0.011294 -0.030152 0.004536 0.009139

... ... ... ... ... ... ... ... ... ... ... ...

22/12/2017 -0.003789 0.000000 0.004195 0.003064 -0.005700 ... -0.003682 -0.002144 0.000098 -0.006377 0.009117

26/12/2017 0.004944 -0.025370 0.014023 -0.004684 0.008544 ... 0.004224 -0.033783 0.000686 0.013142 0.009034

27/12/2017 -0.008515 0.000176 -0.021479 0.003478 -0.006971 ... -0.001052 -0.004370 0.006170 0.007541 0.003304

28/12/2017 0.001145 0.002814 -0.000601 -0.003058 -0.000108 ... -0.002105 0.028494 0.002142 -0.009581 -0.002550

29/12/2017 -0.008197 -0.010814 -0.000201 -0.011044 -0.008316 ... 0.011603 -0.002845 -0.004468 0.022370 0.030248

[993 rows x 483 columns]

* 1. Calculate the principal components of the returns from step (6).

Q7. PCA

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

PCA with n\_components

0 1 2 3 4 5 ... 14 15 16 17 18

19

0 -0.048061 -0.031056 -0.065198 0.041084 0.001070 -0.022087 ... -0.017540 -0.017257 0.097240 0.045030 -0.004236 0.040478

1 -0.024635 -0.029663 -0.011169 -0.008450 -0.011588 -0.015363 ... -0.006762 -0.004439 -0.019051 0.027866 -0.026749 0.011907

2 -0.025896 -0.013753 0.020175 -0.003243 0.004733 -0.006696 ... -0.022194 -0.001648 -0.008977 0.008835 0.032442 -0.063882

3 0.001629 -0.008162 0.038432 -0.011472 -0.010129 0.019024 ... 0.007273 -0.007290 -0.092339 -0.065136 0.108107 -0.008135

4 -0.020461 -0.024251 0.050667 -0.012129 0.003359 -0.018725 ... -0.012695 -0.004949 0.065610 -0.013264 0.019709 -0.040491

.. ... ... ... ... ... ... ... ... ... ... ... ... ...

988 -0.025968 -0.003200 0.006855 -0.013038 -0.010548 -0.012841 ... -0.010428 -0.007480 0.054461 0.127509 -0.008528 0.107027

989 -0.024482 -0.040422 -0.046306 -0.003008 -0.044972 -0.007590 ... -0.024821 0.010626 -0.055814 -0.029391 -0.045164 -0.046363

990 -0.027407 -0.006600 -0.022413 -0.004235 -0.014525 -0.003946 ... 0.010507 0.005942 0.051252 0.012142 -0.054148 -0.026478

991 -0.020963 -0.019046 0.010533 -0.019645 0.014284 0.002361 ... -0.006183 -0.004033 -0.027710 0.009325 -0.011769 -0.013284

992 -0.041827 -0.005160 0.008762 -0.006788 -0.011526 -0.004520 ... -0.028956 0.008517 -0.002148 -0.048737 -0.014186 0.014024

[993 rows x 20 columns]

* 1. Extract the explained variance ratios for the principal components calculated in step (7)
     1. What percentage of variance is explained by the first principal component?
     2. Plot the first 20 explained variance ratios.
     3. Identify an elbow and mark it on the plot.
     4. List your code for this question and provide description of it.

Code:

pca1 = PCA(n\_components=20)

principalComponents = pca1.fit\_transform(df1)

principalDf = pd.DataFrame(data = principalComponents)

df1['Date'] = dt\_array

exp\_var\_pca = pca1.explained\_variance\_ratio\_

cum\_sum\_eigenvalues = np.cumsum(exp\_var\_pca)

print( cum\_sum\_eigenvalues)

plt.figure(figsize=(5,5))

plt.title('First 20 explained variance ratios')

plt.plot(range(0,20), pca1.explained\_variance\_ratio\_)

plt.plot(range(0,20), pca1.explained\_variance\_ratio\_,markevery=(1), ls="", marker="o", label="points")

plt.xlabel('Components')

plt.ylabel('PCA')

plt.figure(figsize=(5,5))

plt.title('First 20 cumulative variance ratios ')

plt.plot( pca1.explained\_variance\_ratio\_,cum\_sum\_eigenvalues)

plt.plot( pca1.explained\_variance\_ratio\_,cum\_sum\_eigenvalues,markevery=(0.5), ls="", marker="o", label="points")

plt.xlabel('PCA')

plt.ylabel('Cumulative variance ratios')

plt.figure(figsize=(5,5))

plt.title('Components vs Cumulative variance ratios ')

plt.plot( range(0,20),cum\_sum\_eigenvalues)

plt.xlabel('Components')

plt.ylabel('Cumulative variance ratios')

Here, in this session, sklearn library is used to import the PCA module. The huge amount of data is projected into 20 dimensions using n\_components from PCA module. The fit.transform method takes the data parameters and applies the transformation in order to get the data is dimensionally reduced. Then the ‘Dates’ is concatenated along axis = 1 for visualization. The explained variance explains the amount of information (variance) that can be attributed to each of the principal components. For that we use explained\_variance\_ratio\_. The np.cumsum finds the cumulative sum of elements on the axis provided.

Once PCA, explained variance and cumulative sum is found, the data is plotted in the graph by providing the data along x and y axis accordingly.

Here 3 graphs are plotted with different x-y combination for data analysis.

From the graphs, elbow point is identified and marked.

Q8. Explained variance ratios for the principal components

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Proportion of Variance Explained :

[0.39641279 0.11989835 0.11870374 0.0374 0.05576135 0.05382723 0.03702429

0.03591661 0.03114627 0.01576371 0.01407073 0.01265708 0.01018959

0.00720374 0.00687813 0.00608469 0.00531079 0.00259806 0.00252078

0.00232272 0.00215395]

Principal components whose eigenvalues are greater than one

[0.69807271 0.21113791 0.20903422 0.09819429 0.09478837 0.06519883

0.06324822 0.05484778 0.02775948 0.02477819 0.02228879 0.01794361

0.01268561 0.01211222 0.01071499 0.00935217 0.00457512 0.00443903

0.00409025 0.00379306]

Percentage of variance explained by the first principal component: 39.64127881035356 %

* 1. Calculate the cumulative variance ratios on the list of explained variance ratios.
     1. Plot all these cumulative variance ratios (x axis = principal component, y axis = cumulative variance ratio).
     2. Mark on your plot the principal component for which the cumulative variance ratio is greater than or equal to 95%.

Q9. Cumulative variance ratios

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

[0.39641279 0.51631114 0.63501488 0.69077622 0.74460346 0.78162775

0.81754436 0.84869063 0.86445433 0.87852506 0.89118214 0.90137173

0.90857548 0.91545361 0.9215383 0.9268491 0.92944716 0.93196794

0.93429066 0.93644461]

* 1. Normalise your dataframe so that the columns have zero mean and unit variance.
     1. Repeat steps (7) - (9) for this new dataframe.
        1. PCA for n components

Q10.1. PCA after standardising

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

PCA with n\_components after standardisation

shape (993, 483)

mean 2.962946571174621e-19

std 1.0

0 1 ... 18 19

0 0.275737 0.065836 ... -4.169722 -0.099830

1 -0.198650 -1.107026 ... -0.119418 -2.385961

2 -0.527002 0.510296 ... 2.378776 -1.277381

3 -1.275877 1.754746 ... 1.713770 0.444899

4 -0.723359 1.025894 ... -1.117769 0.286872

.. ... ... ... ... ...

988 0.880028 -1.896618 ... -0.834271 -0.144865

989 -0.668826 -1.381325 ... -0.439208 0.219942

990 0.466193 -0.403482 ... -1.171239 0.451142

991 0.302051 -0.795504 ... -1.924937 -0.513670

992 -1.086046 0.917010 ... 1.938284 0.144351

[993 rows x 20 columns]

* + - 1. Explained variance ratios for the principal components

Q10.2. Explained variance ratios for the principal components

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Proportion of Variance Explained : [0.06892213 0.06816429 0.04479869 0.03602803 0.03478847 0.03362166

0.03080262 0.02895511 0.0285081 0.02748323 0.02487478 0.02423199

0.02336946 0.02205272 0.01819059 0.01575302 0.01471398 0.01364741

0.01262974 0.00971048]

Principal components whose eigenvalues are greater than one [33.32294831 32.95654052 21.65957867

17.41907885 16.81976731 16.25562965

14.89266338 13.99941709 13.78329362 13.28778109

12.02663159 11.71584918

11.29882992 10.6622005 8.79491237 7.6163787

7.11401631 6.59834433

6.1063137 4.69488921]

Percentage of variance explained by the first principal component: 6.892213344466398 %

* + - 1. Cumulative variance ratios

Q10.3. Cumulative variance ratios

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

[0.06892213 0.13708642 0.18188511 0.21791314 0.2527016 0.28632326

0.31712588 0.34608099 0.37458909 0.40207232 0.4269471 0.45117909

0.47454856 0.49660128 0.51479187 0.53054489 0.54525887 0.55890628

0.57153602 0.5812465 ]

1. **Code**



1. **Full Output**

****