CS 5525

Biplav Choudhury (906089826)

23 September 2020

1 Solutions to Assignment 1

1.1 Problem 1a

Let the two vectors be X and Y where $X = [x_1 \quad x_2 \quad \dots x_N]$ and $Y = [y_1 \quad y_2 \quad \dots y_N]$. Given the averages of X and Y are 0, i.e. $\bar{X} = \bar{Y} = 0$.
The cosine measure of similarity is given by

$$cos(X, Y) = \frac{X.Y}{|X||Y|} \tag{1}$$

where $|X|$ and $|Y|$ are the vector lengths.

$$X.Y = x_1y_1 + x_2y_2 + \dots x_ny_n \tag{2}$$

$$|X| = \sqrt{{x_1}^2 + {x_2}^2 + \dots {x_n}^2} \tag{3}$$

$$|Y| = \sqrt{{y_1}^2 + {y_2}^2 + \dots {y_n}^2} \tag{4}$$

Therefore the cosine similarity measure becomes

$$cos(X, Y) = \frac{x_1y_1 + x_2y_2 + \dots x_ny_n}{\sqrt{{x_1}^2 + {x_2}^2 + \dots {x_n}^2}.\sqrt{{y_1}^2 + {y_2}^2 + \dots {y_n}^2}} \tag{5}$$

Now for the correlation, it is given by

$$corr(X, Y) = \frac{cov(X, Y)}{S_xS_y} \tag{6}$$

where $cov(X, Y)$ is the covariance between X and Y, while S_x, S_y are the standard deviations of X and Y respectively. For $\bar{X} = \bar{Y} = 0$, they are given by

$$cov(X, Y) = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \bar{X})(y_i - \bar{Y}) = \frac{1}{N-1}\sum_{i=1}^{N}(x_iy_i) = \frac{1}{N-1}(x_1y_1 + x_2y_2 + \dots + x_ny_n) \tag{7}$$

$$S_x = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \bar{X})^2} = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i)^2} = \sqrt{\frac{1}{N-1}(x_1^2 + x_2^2 + \dots + x_n^2)} \tag{8}$$

$$S_y = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(y_i - \bar{Y})^2} = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(y_i)^2} = \sqrt{\frac{1}{N-1}(y_1^2 + y_2^2 + \cdots + y_n^2)} \tag{9}$$

Therefore from Eqns. 7, 8 and 9, the correlation becomes

$$corr(X,Y) = \frac{x_1y_1 + x_2y_2 + \ldots x_ny_n}{\sqrt{x_1{}^2 + x_2{}^2 + \ldots x_n{}^2}.\sqrt{y_1{}^2 + y_2{}^2 + \ldots y_n{}^2}} \tag{10}$$

which is the same as Eqn. 5. Therefore for the current conditions, the cosine similarity measure and the correlation are the same

1.2 Problem 1b

For two vectors X and Y, euclidean distance is given by

$$D = \sqrt{(X-Y)^2} = \sqrt{(x_1-y_1)^2 + (x_2-y_2)^2 \cdots + (x_n-y_n)^2} \tag{11}$$

$$= \sqrt{(x_1^2 + x_2^2 + \cdots + x_n^2) + (y_1^2 + y_2^2 + \cdots + y_n^2) - 2(x_1y_1 + x_2y_2 + \cdots + x_ny_n)} \tag{12}$$

The cosine similarity is given by

$$cos(X,Y) = \frac{X.Y}{\sqrt{|X||Y|}} = X.Y = x_1y_1 + x_2y_2 + \ldots x_ny_n \tag{13}$$

where we have used $|X| = |Y| = 1$. Using Eqn. 13 in Eqn. 12,

$$D = \sqrt{1 + 1 - 2cos(X,Y)} = \sqrt{2(1 - cos(X,Y))} \tag{14}$$

1.3 Problem 2

Based on the idea of $tf - idf$, we have to assign weights $C_{w_i}^D$ and $E_{w_i}^D$ that are are able to quantify the ability of word w_i in representing domain D and distinguishing event e from other events in the same domain.

$C_{w_i}^D$ - Building on the construction of $tf - idf$, we will have two terms : first one is tf_D and will see the normalized frequency of the word w_i occurring among all the articles in representing the target domain D, and represents how important the word is to that domain. The second term idf_D will ensure words that are more specific to D are given higher weightage.

$$C_{w_i}^D = tf_D \times idf_D = \frac{\sum_{A \in A_D} f(w,A)}{max(\sum_{A \in A_D f(w,A)})} \times \frac{N}{N_D} \tag{15}$$

where N_D is the number of times the word W_i appears in A_D while N is the total number of documents.

$E_{w_i}^e$ - Building on the construction of $tf - idf$, we will have two terms : first one is tf_D^E and will see the normalized frequency of the word w_i occurring among all the articles that represent the event e inside domain D, and represents how important the word is to that event inside the domain. The second term idf_D^e will ensure words that are more specific to e inside D are given higher weightage. Note that the domain is fixed in this case and all the documents involved in the calculation belong to the target domain D.

$$E_{w_i}^e = tf_D^e \times idf_D^e = \frac{\sum_{A \in A_e, e \in D} f(w, A)}{max(\sum_{A \in A_e, e \in D f(w,A)})} \times \frac{N^D}{N_D^e} \tag{16}$$

where N_D^e is the number of times the word w_i appears in A_e.

Also the concept of Latent Semantic Indexing (LSI) might be used here by constructing the domains as concepts from concept space, but in this question as the domains are already provided, LSI might not be needed directly.

1.4 Problem 3

Let the location of Mike be (x_M, y_M) and location of John be (x_J, y_J) in a 2D plane, and the equation of the circle is $(x_M - x_J)^2 + (y_M - y_J)^2 = r^2$ where r is the radius of the circular track.

- Euclidean Distance : $D = \sqrt{(x_M - x_J)^2 + (y_M - y_J)^2}$. As Euclidean distance is the length of the line connecting the two points, it is constant between Mike and John. So $D_{max} = D_{min} = r$

- Manhattan Distance : $D = |x_M - x_J| + |y_M - y_J|$. In this case, let $x_M - x_J = \Delta x, y_M - y_J = \Delta y$ and $D = |\Delta x| + |\Delta y|$.

  As the square of a real number is greater or equal to 0,

  $$\begin{aligned} |\Delta x - \Delta y|^2 &\geq 0 \\ \implies (\Delta x)^2 + (\Delta y)^2 - 2(\Delta x \Delta y) &\geq 0 \end{aligned}$$

  Add $(|\Delta x| + |\Delta y|)^2$ on both sides.

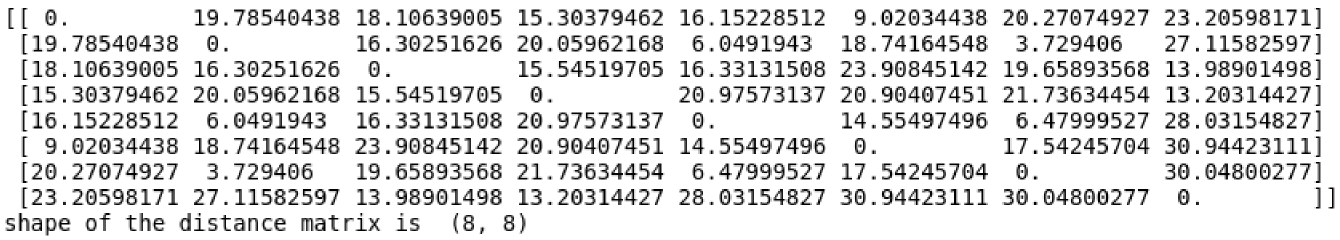
  $$\begin{aligned} \implies 2(\Delta x)^2 + 2(\Delta y)^2 &\geq (|\Delta x| + |\Delta y|)^2 \\ \implies \sqrt{2} \times \text{Euclidean Distance} &\geq \text{Manhattan Distance} \end{aligned}$$

  The upper limit of Manhattan distance is bounded by the Euclidean distance, hence $D_{max} = \sqrt{2}r$. Regarding D_{min}, I was not able to do it mathematically but by visual inspection, it seems it will be least when either $\Delta x = 0$ or Δy=0, in which case $D_{min} = r$.

- Chebyshev Distance : $D = max(|x_J - x_M|, |y_J - y_M|)$. In this case, let $x_M - x_J = \Delta x, y_M - y_J = \Delta y$ and $D = max(|\Delta x|, |\Delta y|)$. Due to the equation of the circle, $(\Delta x)^2 + (\Delta y)^2 = r^2$, therefore Δx is maximum and equals r when $\Delta y = 0$ and vice versa. Hence $D_{max} = r$. D_{min} occurs when $\Delta x = \Delta y$ which makes $D_{min} = \frac{r}{\sqrt{2}}$.

1.5 Problem 4 part 1

The distance matrix for the first 8 points are :



```
[[ 0.         19.78540438 18.10639005 15.30379462 16.15228512  9.02034438 20.27074927 23.20598171]
 [19.78540438  0.         16.30251626 20.05962168  6.0491943  18.74164548  3.729406   27.11582597]
 [18.10639005 16.30251626  0.         15.54519705  0.         20.97573137 20.90407451 21.73634454 13.20314427]
 [15.30379462 20.05962168 15.54519705  0.         20.97573137 20.90407451 21.73634454 13.20314427]
 [16.15228512  6.0491943  16.33131508 20.97573137  0.         14.55497496  6.47999527 28.03154827]
 [ 9.02034438 18.74164548 23.90845142 20.90407451 14.55497496  0.         17.54245704 30.94423111]
 [20.27074927  3.729406   19.65893568 21.73634454  6.47999527 17.54245704  0.         30.04800277]
 [23.20598171 27.11582597 13.98901498 13.20314427 28.03154827 30.94423111 30.04800277  0.        ]]
shape of the distance matrix is  (8, 8)
```

Figure 1: Distance matrix for the first 8 points

1.6 Problem 4 part 1a

The nearest neighbor based graph for the 200 points was plotted in network_x and it looks like

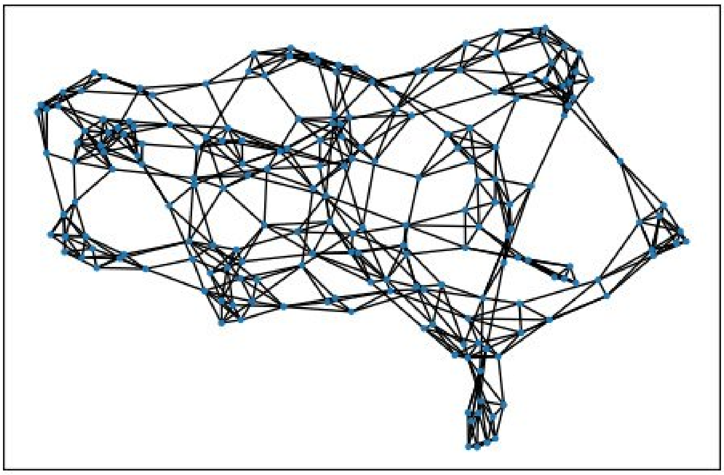


Figure 2: Graph based on nearest neighbor

1.7 Problem 4 part 1b

The radius based graph for the 200 points was plotted in network_x and it looks like

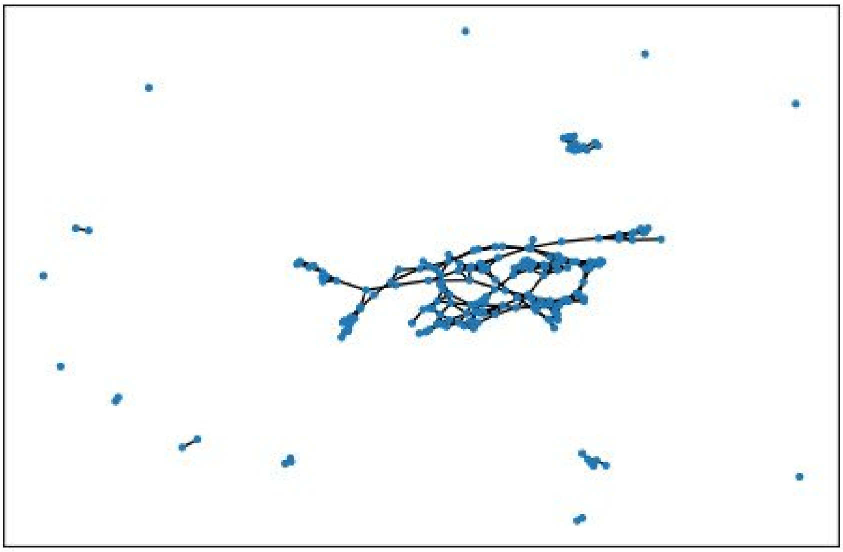
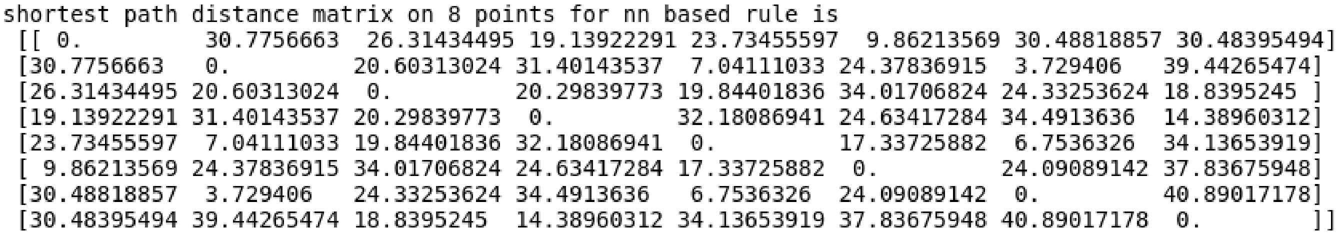


Figure 3: Graph based on radius

1.8 Problem 4 part 2

The 200*200 distance matrix based on the geodesic distance was calculated, Dijkstra shortest path applied and the 8*8 part of it has been reproduced below.

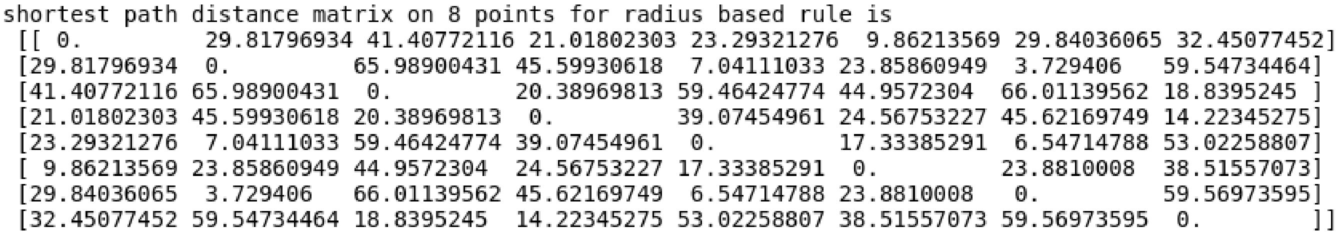
The nearest neighbor based one is



```
shortest path distance matrix on 8 points for nn based rule is
 [[ 0.         30.7756663  26.31434495 19.13922291 23.73455597  9.86213569 30.48818857 30.48395494]
 [30.7756663   0.         20.60313024 31.40143537  7.04111033 24.37836915  3.729406   39.44265474]
 [26.31434495 20.60313024  0.         20.29839773 19.84401836 34.01706824 24.33253624 18.8395245 ]
 [19.13922291 31.40143537 20.29839773  0.         32.18086941 24.63417284 34.4913636  14.38960312]
 [23.73455597  7.04111033 19.84401836 32.18086941  0.         17.33725882  6.7536326  34.13653919]
 [ 9.86213569 24.37836915 34.01706824 24.63417284 17.33725882  0.         24.09089142 37.83675948]
 [30.48818857  3.729406   24.33253624 34.4913636   6.7536326  24.09089142  0.         40.89017178]
 [30.48395494 39.44265474 18.8395245  14.38960312 34.13653919 37.83675948 40.89017178  0.        ]]
```

Figure 4: Dijkstra shortest path

The radius neighbor based one is



```
shortest path distance matrix on 8 points for radius based rule is
 [[ 0.         29.81796934 41.40772116 21.01802303 23.29321276  9.86213569 29.84036065 32.45077452]
 [29.81796934  0.         65.98900431 45.59930618  7.04111033 23.85860949  3.729406   59.54734464]
 [41.40772116 65.98900431  0.         20.38969813 59.46424774 44.9572304  66.01139562 18.8395245 ]
 [21.01802303 45.59930618 20.38969813  0.         39.07454961 24.56753227 45.62169749 14.22345275]
 [23.29321276  7.04111033 59.46424774 39.07454961  0.         17.33385291  6.54714788 53.02258807]
 [ 9.86213569 23.85860949 44.9572304  24.56753227 17.33385291  0.         23.8810008  38.51557073]
 [29.84036065  3.729406   66.01139562 45.62169749  6.54714788 23.8810008   0.         59.56973595]
 [32.45077452 59.54734464 18.8395245  14.22345275 53.02258807 38.51557073 59.56973595  0.        ]]
```

Figure 5: Dijkstra shortest path

1.9 Problem 5 part 1

The box plot looks like

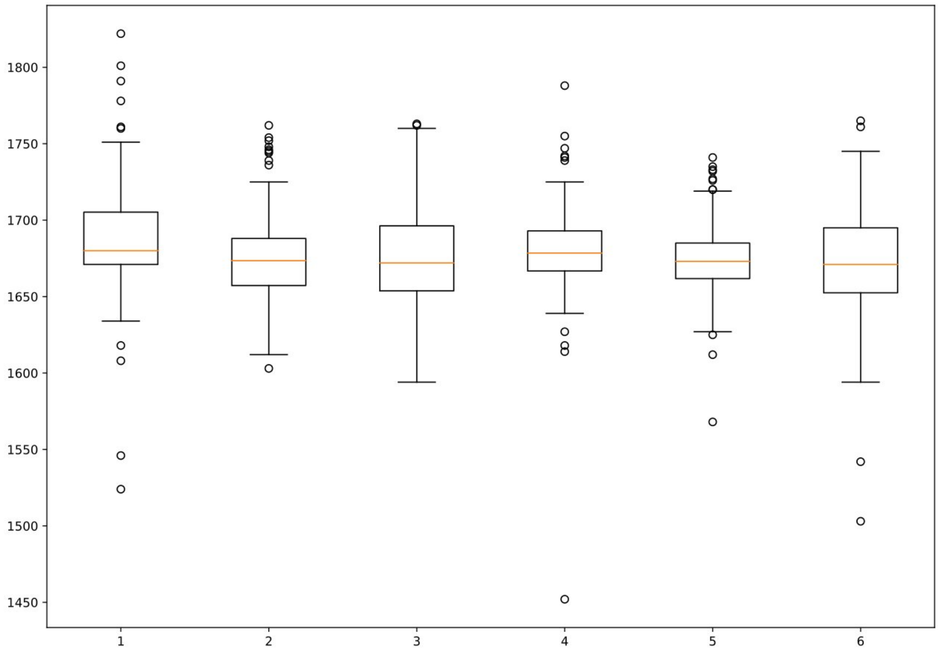


Figure 6: Box Plot

1.10 Problem 5 part 2

The red center line is the median while the outer edges of the box represent the 25th and 75th percentile.

One observation is that the median is not exactly in between the two percentiles - it is closer to the 25th percentile for Pos1, Pos4 and Pos6 while it is is slightly closer to the 75th percentile for Pos2.

Asymmetric box-plots mean that the data is skewed. For example for Pos1, the boxplot tells us that the values between the 25th percentile and the median are closer to each other while the values between the median and the 75th percentile are spread apart, which makes the data skewed towards the higher side.

1.11 Problem 6 part 3

The data after preprocessing belong in the same range whereas before each feature had its own range.

1.12 Problem 6 part 4

Correlation between two variables x and y is given by

$$correlation = \frac{S_{xy}}{S_x S_y}$$

where S_{xy} could be the sample/population covariance between x and y, and S_x and S_y are the standard deviations in x and y respectively.

The next parts until part 12 has been implemented in the python notebook.

1.13 Problem 6 part 12

Answer is motivated by this post on mathexchange.
Assume that the data matrix X with n samples has been preprossessed to have zero mean, which is compulsorily done with PCA. This also makes the eigen vectors orthonormal. Now the covariance matrix is calculated as $\frac{XX^T}{n-1}$. As the covariance matrix is symmetric, it is diagonizable.

$$\frac{XX^T}{n-1} = \frac{WDW^T}{n-1}$$

where D is a diagonal matrix. Then applying SVD to X

$$X = USV^T \tag{17}$$

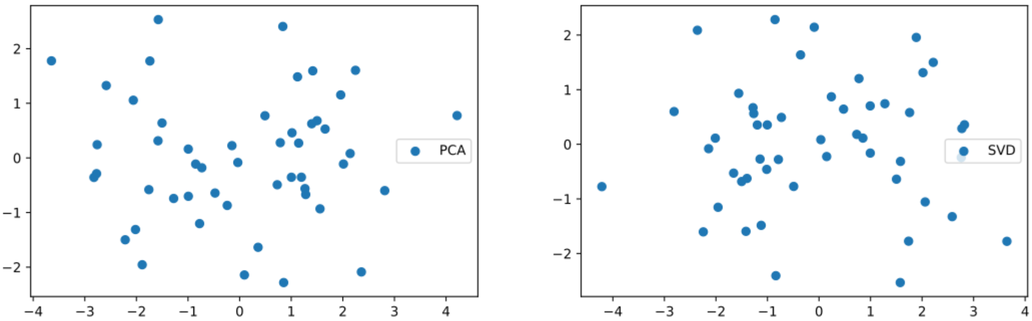
We will use Eqn. 17 to construct the covariance matrix for X

$$\frac{XX^T}{n-1} = \frac{(USV^T)(USV^T)^T}{n-1} = \frac{(USV^T)(VSU^T)}{n-1} \tag{18}$$

In SVD, V represents the eigen vectors which are orthonormal, i.e. $V^TV = I$ and Eqn. 18 becomes

$$\frac{XX^T}{n-1} = \frac{US^2U^T}{n-1} \tag{19}$$

The final expression show that there is a very strong relationship between SVD and PCA, and the principal components can be calculated directly using SVD rather than by the covariance and eigen vector approach.



(a) PCA in 2D

(b) SVD in 2D

Figure 7: PCA and SVD 2D projection comparision

The images look very similar and when I rotate the image by 180 degrees, they become identical. Below is the image of the SVD plot rotated by 180 degrees. The rotation was done in latex so tha x-ticks and y-ticks look like that.

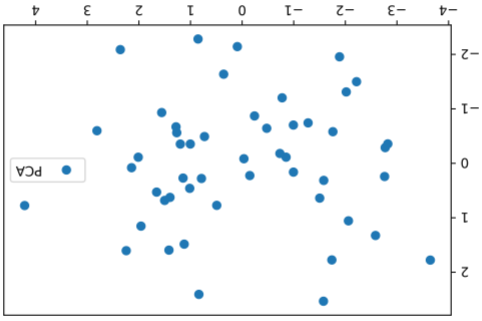


Figure 8: PCA rotated 180 degrees