Data Science and Social Inquiry: HW2

B99303081 Yu-Chang Chen

November 7, 2023

Question 1: Matrix operations

(a)

$$\begin{aligned} E(\boldsymbol{Y}) &= E(\boldsymbol{AX}) + E(\boldsymbol{b}) \\ &= \boldsymbol{A}E(\boldsymbol{X}) + \boldsymbol{b} \\ &= \boldsymbol{A\mu} + \boldsymbol{b} \end{aligned}$$

(b)

$$\begin{aligned} \text{Cov}(\boldsymbol{Y}) &= E[(\boldsymbol{Y} - \boldsymbol{E}(\boldsymbol{Y}))(\boldsymbol{Y} - \boldsymbol{E}(\boldsymbol{Y}))^T] \\ &= E[(\boldsymbol{AX} + \boldsymbol{b} - \boldsymbol{A\mu} - \boldsymbol{b})(\boldsymbol{AX} + \boldsymbol{b} - \boldsymbol{A\mu} - \boldsymbol{b})^T] \\ &= E[(\boldsymbol{AX} - \boldsymbol{A\mu})(\boldsymbol{AX} - \boldsymbol{A\mu})^T] \\ &= \boldsymbol{A}E[(\boldsymbol{X} - \boldsymbol{\mu})(\boldsymbol{X} - \boldsymbol{\mu})^T]\boldsymbol{A}^T \\ &= \boldsymbol{A\Sigma A^T} \end{aligned}$$

Question 2: PCA with non-diagonal covariance matrix

(c) What is the first principal component? Explain how you reach your answer carefully and write down its coefficients.

Sol.

Originally, we want to solve

$$max\ a_i^{'}\Sigma a_i\ s.t.\ a_i^{'}a_i = 1$$

We know $\Sigma = PDP^{-1} = PDP^{'}$

$$a_i^{'}\Sigma a_i = a_i^{'}PDP^{'}a_i = (P^{'}a_i)^{'}D(P^{'}a_i) = b_i^{'}Db_i$$

$$a_i^{'}a_i = a_i^{'}(PP^{-1})a_i = a_i^{'}(PP^{'})a_i = (P^{'}a_i)^{'}(P^{'}a_i) = b_i^{'}b_i$$

We can rewrite our optimization problem as

$$max\ b_i^{'}Db_i\ s.t.\ b_i^{'}b_i = 1$$

$$\mathbf{b}_i = \begin{pmatrix} b_{i1} & b_{i2} & b_{i3} & b_{i4} & b_{i5} \end{pmatrix}$$

Therefore, the optimization problem will be

$max\ 0.855b_{i1}^2 + 0.942b_{i2}^2 + 0.738b_{i3}^2 + 0.109b_{i4}^2 + 2.024b_{i5}^2 \quad s.t.\ b_{i1}^2 + b_{i2}^2 + b_{i3}^2 + b_{i4}^2 + b_{i5}^2 = 1$

Solve the optimization problem for PC_1

$max\ 0.855b_{11}^2 + 0.942b_{12}^2 + 0.738b_{13}^2 + 0.109b_{14}^2 + 2.024b_{15}^2 \quad s.t.\ b_{11}^2 + b_{12}^2 + b_{13}^2 + b_{14}^2 + b_{15}^2 = 1$

Thus,

$$b_{15}^2 = 1, b_{11} = b_{12} = b_{13} = b_{14} = 0$$

$$b_{15} = \pm 1$$

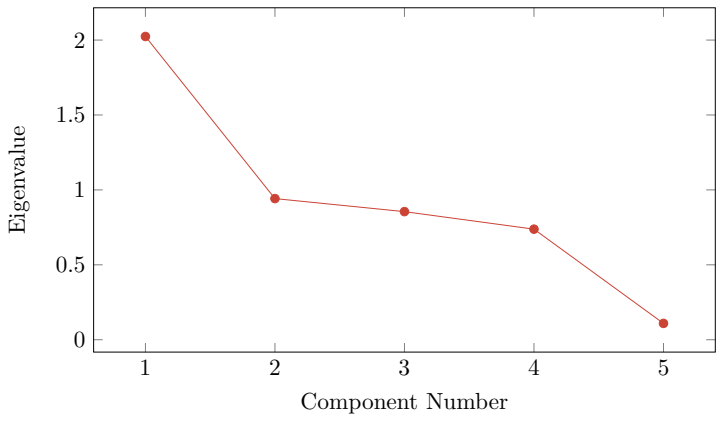
$$b_1 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \; a_1 = P \cdot b_1 = \begin{pmatrix} 0.006 \\ 0.268 \\ -0.015 \\ -0.798 \\ 0.539 \end{pmatrix}$$

$$PC1 = a_1 X = 0.006X_1 + 0.268X_2 - 0.015X_3 - 0.798X_4 + 0.539X_5$$

(*d*) Find the corresponding variance of each principal component. Graph the scree plot for the five components.

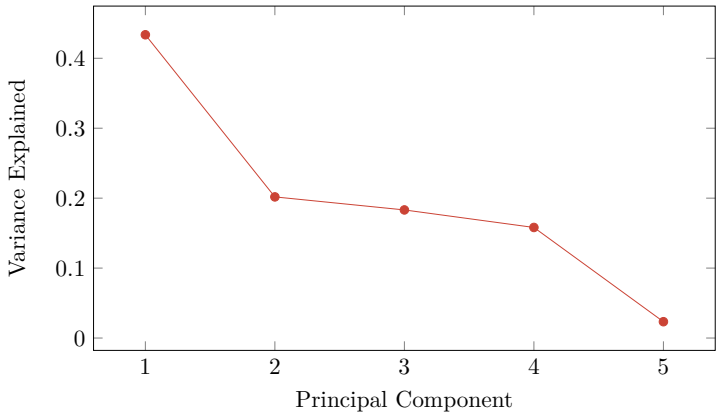
Sol. The variance explained by each component:

$$\begin{aligned} \text{for} \quad & PC_1 : \; Var(PC_1) = 2.024 \\ \text{for} \quad & PC_2 : \; Var(PC_2) = 0.942 \\ \text{for} \quad & PC_3 : \; Var(PC_3) = 0.855 \\ \text{for} \quad & PC_4 : \; Var(PC_4) = 0.738 \\ \text{for} \quad & PC_5 : \; Var(PC_5) = 0.109 \end{aligned}$$



The proportion of variance explained by each component:

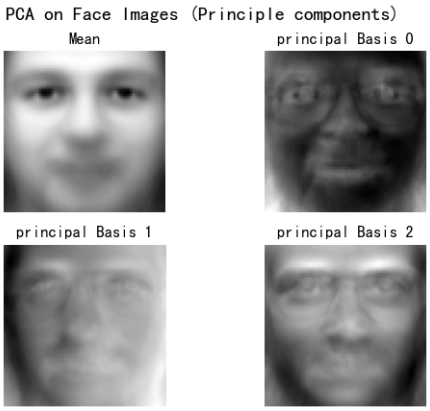
$$for PC_1: \frac{Var(PC_1)}{\sum_{i=1}^{5} Var(PC_i)} = \frac{2.024}{4.668} = 0.4335904$$
$$for PC_2: \frac{Var(PC_2)}{\sum_{i=1}^{5} Var(PC_i)} = \frac{0.942}{4.668} = 0.20179949$$
$$for PC_3: \frac{Var(PC_3)}{\sum_{i=1}^{5} Var(PC_i)} = \frac{0.855}{4.668} = 0.18316195$$
$$for PC_4: \frac{Var(PC_4)}{\sum_{i=1}^{5} Var(PC_i)} = \frac{0.738}{4.668} = 0.15809769$$
$$for PC_5: \frac{Var(PC_5)}{\sum_{i=1}^{5} Var(PC_i)} = \frac{0.109}{4.668} = 0.02335047$$



Question 3: Image Compression with PCA

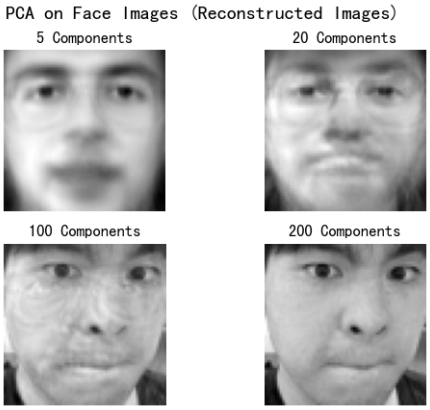
(*e*) Find the principal components, referred to as the " principal faces" . Plot both the mean face and the first three principal faces.

Sol.



(f) Focus on compressing the last face present in our dataset. Achieve this com- pression by computing inner products with the principal faces. Subsequently, plot the reconstructed faces using q = 5, 20, 100, 200 principal components.

Sol.



(g) When employing q = 200 principal components for data compression, what percentage of storage space do we save compared to the size of the original dataset?

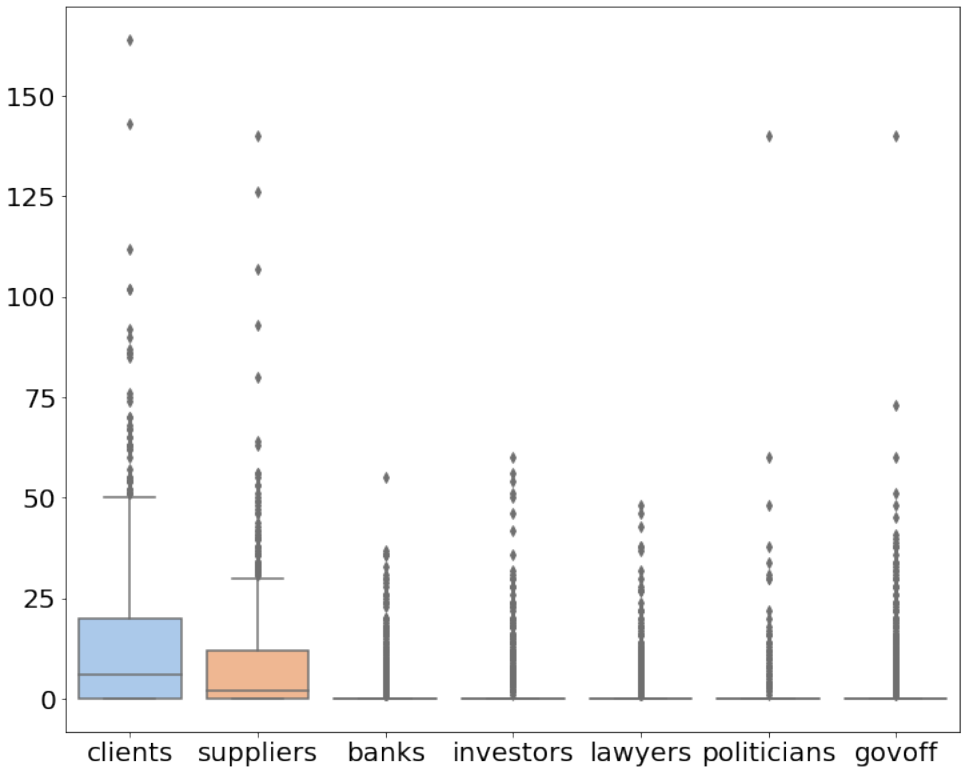
Sol.

$$\begin{aligned}
\text{Percentage saved} &= \frac{\text{original size} - \text{compressed size}}{\text{original size}} \times 100\% \\
&= \frac{N \times D - N \times 200}{N \times D} \times 100\% \\
&= \frac{D - 200}{D} \times 100\% \\
&= \frac{4096 - 200}{4096} \times 100\% \\
&= 95.117\%
\end{aligned}$$

Question 4: Prototyping CEO's behavior

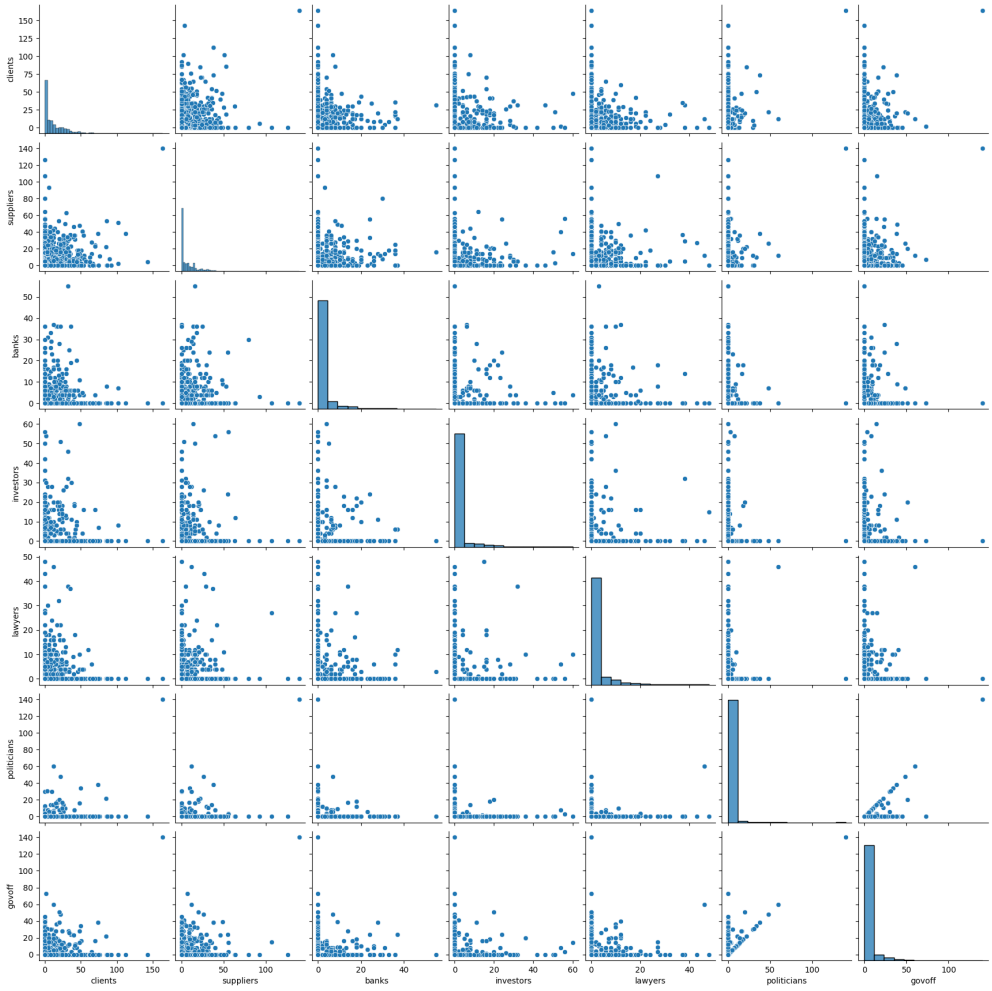
(h) Initiate every data analysis by scrutinizing the raw data. Utilize a box plot to summarize the seven marginal distributions.

Sol.



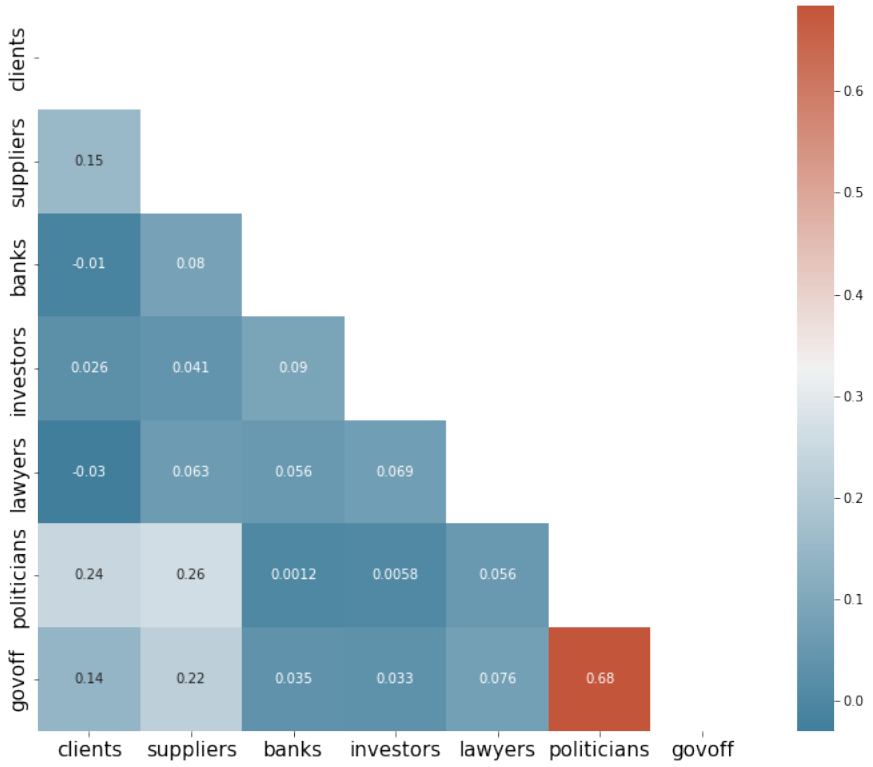
(i) Subsequent to the box plot, construct a pair plot for these seven variables. What observations can be made from the box plot and pair plot?

Sol.



(j) Use a heatmap to summarize the correlations between the number of activities. Which type correlates with type **politicians** most?

Sol. Government officials correlates with type politicians the most.



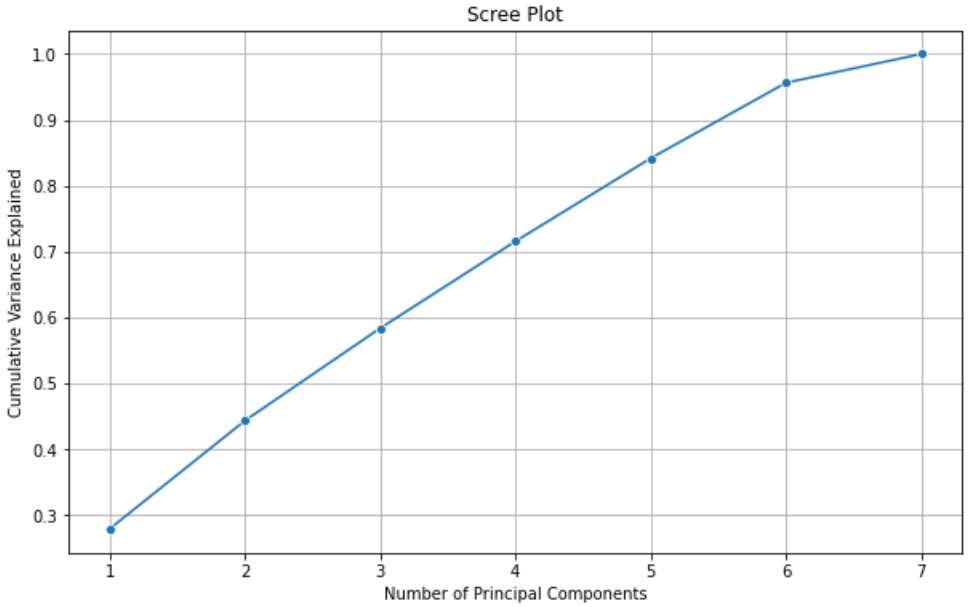
(k) Standardize the seven variables by centering around their mean and scaling by their standard deviations. Execute PCA to determine the first principal component.

Sol.

clients	suppliers	banks	investors	lawyers	politicians	govoff
0.3072731	0.37774657	0.06364262	0.0638176	0.10872277	0.62184173	0.59687502

(l) (1 pt) Make the scree plot. How many principal components are needed to explain 70% of the variation?

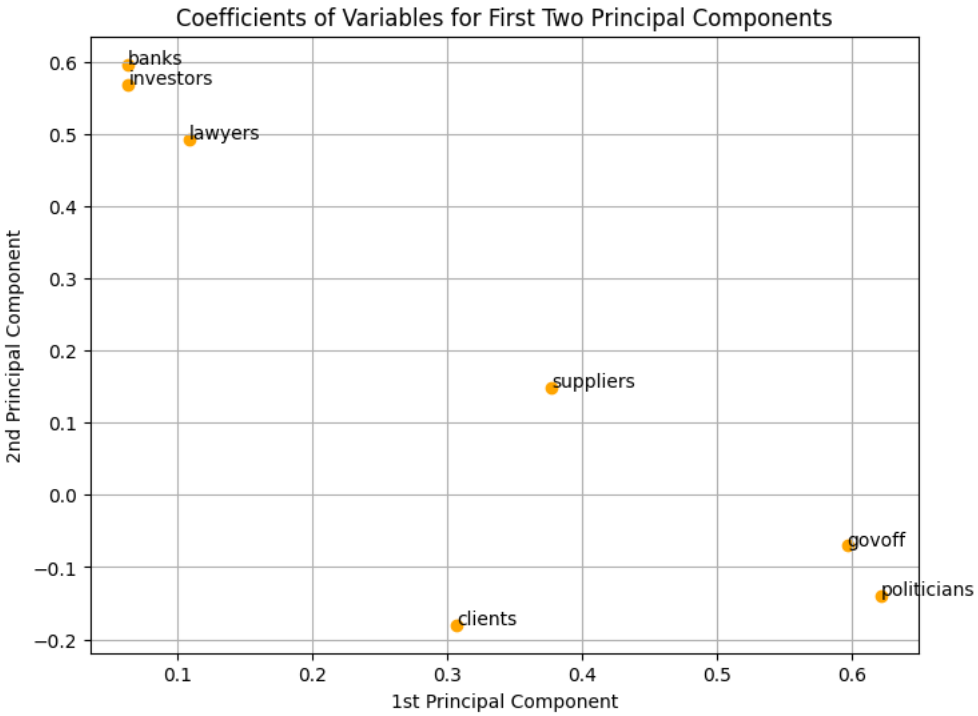
Sol.



4 PCs are needed to explain 70% of the variation.

(m) (1 pt) Put the first component on the x-axis and the second component on the y-axis. Plot the coefficients of each variable. How would you interpret the first two components?

Sol.



Question 5: Practicing the hierarchical clustering algorithm

(n) Perform the hierarchical clustering with average linkage. Clearly indicate which observations are pooled in each step.

Sol. Each pairwise squared distance:

	$(0,4)$	$(-3,1)$	$(3,3)$	$(3,5)$	$(-3,3)$
$(0,4)$		18	10	10	10
$(-3,1)$			40	52	4
$(3,3)$				4	36
$(3,5)$					40
$(-3,3)$					

So

$$\{(0,4)\},\{(-3,1)\},\{(3,3)\},\{(3,5)\},\{(-3,3)\} \Rightarrow \{0,4\},\{(3,3),(3,5)\},\{(-3,1),(-3,3)\}.$$

Next,

$$\begin{aligned} d(\{(0,4)\},\{(3,3),(3,5)\}) &= \frac{10+10}{2} = 10 \\ d(\{(0,4)\},\{(-3,1),(-3,3)\}) &= \frac{18+10}{2} = 19 \\ d(\{(3,3),(3,5)\},\{(-3,1),(-3,3)\}) &= \frac{40+36+52+40}{4} = 42, \end{aligned}$$

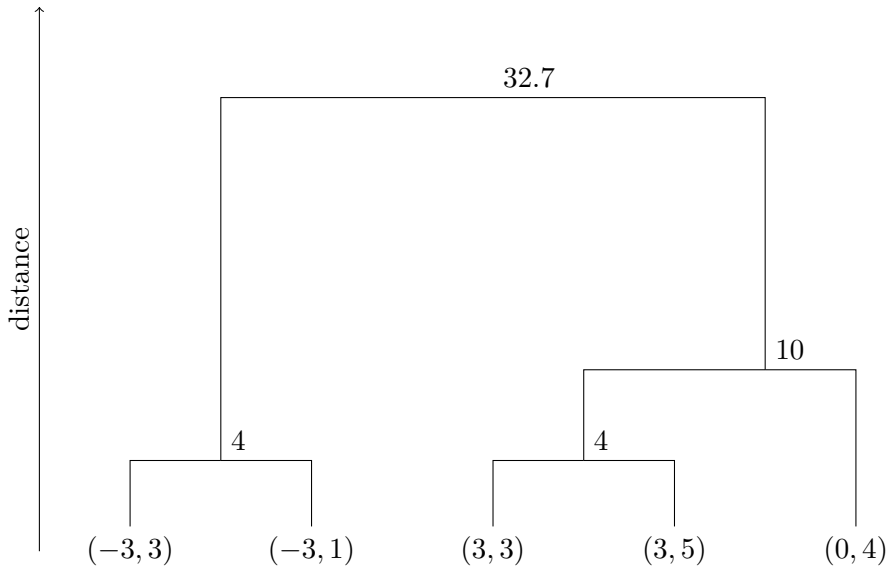
so

$$\{(0,4)\},\{(3,3),(3,5)\},\{(-3,1),(-3,3)\} \Rightarrow \{(0,4),(3,3),(3,5)\},\{(-3,1),(-3,3)\},$$

and finally,

$$d(\{(0,4),(3,3),(3,5)\},\{(-3,1),(-3,3)\}) = \frac{18+10+40+36+52+40}{6} = 32.7.$$

The dendogram of the clustering can be shown as



(*o*) Perform the hierarchical clustering with single linkage.

Sol. Similarly, we first group the closest points,

$$\{(0,4)\},\{(-3,1)\},\{(3,3)\},\{(3,5)\},\{(-3,3)\} \Rightarrow \{0,4\},\{(3,3),(3,5)\},\{(-3,1),(-3,3)\}.$$

Next

$$\begin{aligned} &d(\{(0,4)\},\{(3,3),(3,5)\}) = 10 \\ &d(\{(0,4)\},\{(-3,1),(-3,3)\}) = 18 \\ &d(\{(3,3),(3,5)\},\{(-3,1),(-3,3)\}) = 52, \end{aligned}$$

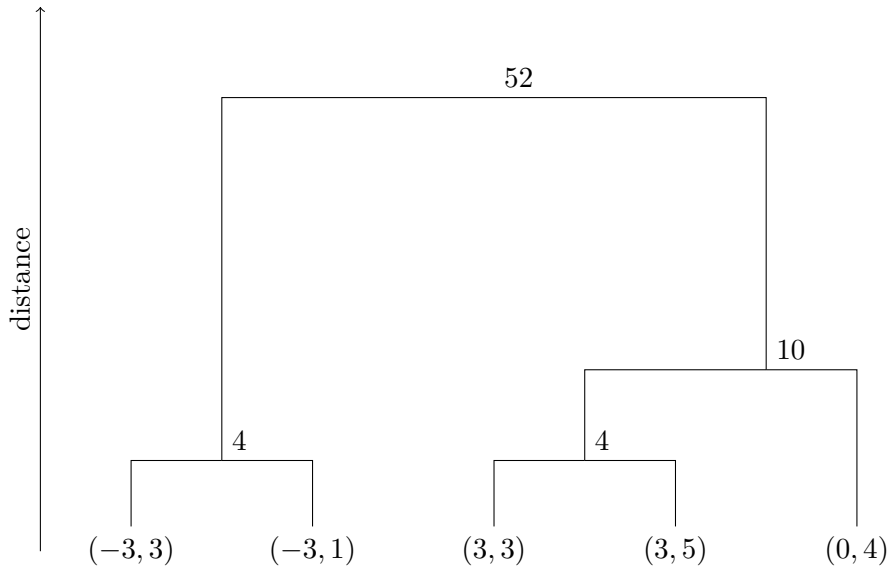
so

$$\{0,4\},\{(3,3),(3,5)\},\{(-3,1),(-3,3)\} \Rightarrow \{(0,4),(3,3),(3,5)\},\{(-3,1),(-3,3)\},$$

and finally,

$$d(\{(0,4),(3,3),(3,5)\},\{(-3,1),(-3,3)\}) = 52$$

The dendogram of the clustering can be shown as



References Code

1. reference code for Q3

```
import os
import warnings
from numpy.linalg import matrix_rank
from matplotlib import pyplot as plt
import numpy as np
from sklearn.decomposition import PCA
# This block of code help you read and show the data
# Nothing need to be done here
# Set seed
np.random.seed(5516)
# Read the faces. The data set is a modified version from the Olivetti data
set.add
# For a description for the original data set,
# see https://scikit-learn.org/stable/modules/generated/sklearn.datasets.fetch_olivett
X = np.loadtxt('/Users/ycchen/Desktop/test/faces.csv', delimiter=',')

# height, width, and number of images
h, w, n = 64, 64, 401
```

```
# Pick 16 random faces to show
val = np.random.choice(n, 16, replace=False)
fig, axs = plt.subplots(4, 4)
fig.suptitle("16 Random Face Images ", fontsize="x-large")
for i in range(16):
r, c = int(i / 4), i % 4
axs[r, c].imshow(X[val[i]].reshape(h, w), cmap='gray')
axs[r, c].axis('off')
# Question 1: Plot the mean faces and the first principal components ("principal
faces")
# Remember to center the data and before PCA
# PS. no need to rescale

# Your task find mu, XC and A:
# mu = mean face. Should be 4096 x 1
# XC = X - mu, the centered X
# A = the matrix with principal components as columns. Should be 4096 x
401
# Recall that A has dimension p x q.
# Here q = 401 column because we only have 401 faces in our data.
# If we have more than 4096 faces (the number of variables, q = 4096 = p)
# and the following code help you plot the result.
mu = np.mean(X, axis=0)
XC = X - mu
pca = PCA()
pca.fit(XC)
A = pca.components_.T

# The following code help you plot the result
# mu = mean face
# V = the matrix with principal components as columns
fig, axs = plt.subplots(2, 2)
fig.suptitle("PCA on Face Images (Principle components) ", fontsize="x-large")
for i in range(4):
r, c = int(i / 2), i % 2
if r == 0 and c == 0:
axs[r, c].imshow(mu.reshape(h, w), cmap='gray')
axs[r, c].axis('off')
axs[r, c].set_title('Mean')
else:
```

```
axs[r, c].imshow(A.T[i - 1].reshape(h, w), cmap='gray')
axs[r, c].axis('off')
axs[r, c].set_title('principal Basis {}'.format(i - 1))
```

2. reference code for Q4

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.decomposition import PCA

df = pd.read_csv('survey_response_data.csv')
interest = ['clients', 'suppliers', 'banks', 'investors', 'lawyers', 'politicians',
'govoff']

agg = df.groupby(['id'])[interest].sum()
agg.head()

sns.set_palette("pastel")
f, ax = plt.subplots(figsize=(12, 10))
g = sns.boxplot(data=agg)
g.tick_params(labelsize=20)
fig = g.get_figure()
fig.savefig("box.pdf")
corr = agg.corr()
f, ax = plt.subplots(figsize=(12, 10))

sns.pairplot(df)
plt.show()

mask = np.triu(np.ones_like(corr, dtype=bool))

cmap = sns.diverging_palette(230, 20, as_cmap=True)

g = sns.heatmap(corr, annot=True, mask=mask, cmap=cmap)
g.tick_params(labelsize=15)
fig = g.get_figure()
fig.savefig("heat.pdf")
```

```
scaler = StandardScaler()
agg_std = scaler.fit_transform(agg)

pca = PCA(n_components=1)
pca.fit(agg_std)
first_principal_component = pca.components_[0]
print("the first principal component: ", first_principal_component)

V = pca.components_
W = pca.explained_variance_

def outputLatex(V=V[0,:],l=interest):
txt = ''
for i in range(len(l)):
txt += str(round(V[i],2)) + '\text{' + l[i] + '}'
return txt

outputLatex()
```