컴퓨터 통계 방법론 HW5

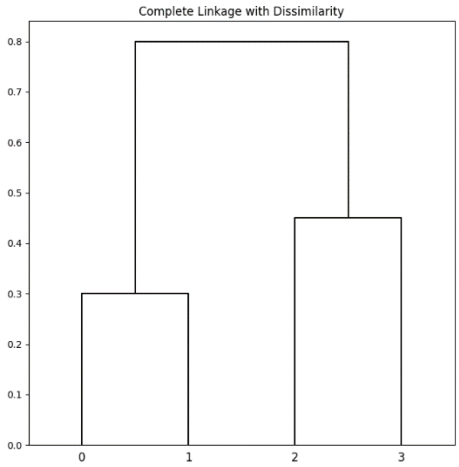
20180490 이재헌

1. Ch12-Prob2: Suppose that we have four observations, for which we compute a dissimilarity matrix, given by

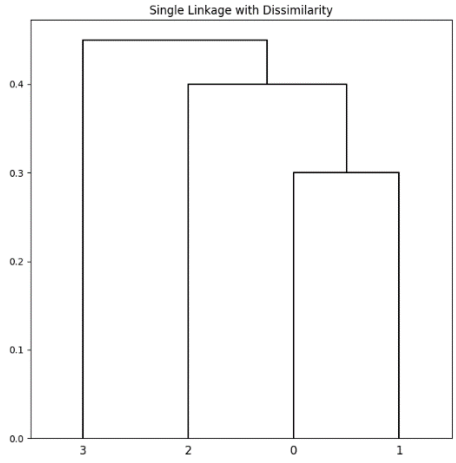
$$\begin{bmatrix} & 0.3 & 0.4 & 0.7 \\ 0.3 & & 0.5 & 0.8 \\ 0.4 & 0.5 & & 0.45 \\ 0.7 & 0.8 & 0.45 & \end{bmatrix}.$$

For instance, the dissimilarity between the first and second observations is 0.3, and the dissimilarity between the second and fourth observations is 0.8

(a) On the basis of this dissimilarity matrix, sketch the dendrogram hat results from hierarchically clustering these four observations using complete linkage. Be sure to indicate on the plot the height at which each fusion occurs, as well as the observations corresponding to each leaf in the dendrogram.



(b) Repeat (a), this time using single linkage clustering.



(c) Suppose that we cut the dendrogram obtained in (a) such that two clusters result. Which observations are in each cluster?

```
cut_tree(linkage_comp, n_clusters=2).T
```

```
array([[0, 0, 1, 1]])
```

0 & 1 form one cluster and 2 & 3 form the other cluster.

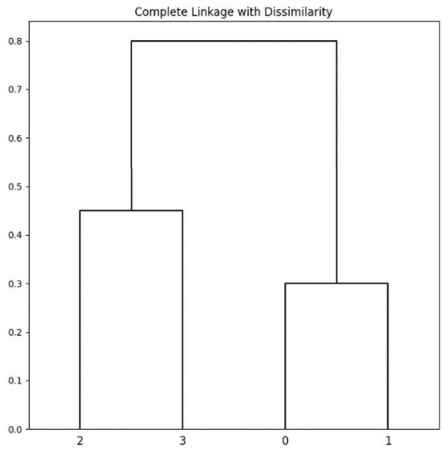
(d) Suppose that we cut the dendrogram obtained in (b) such that two clusters result. Which observations are in each cluster?

```
cut_tree(linkage_sing, n_clusters=2).T
```

```
array([[0, 0, 0, 1]])
```

0, 1 & 2 form one cluster and 3 forms the other cluster.

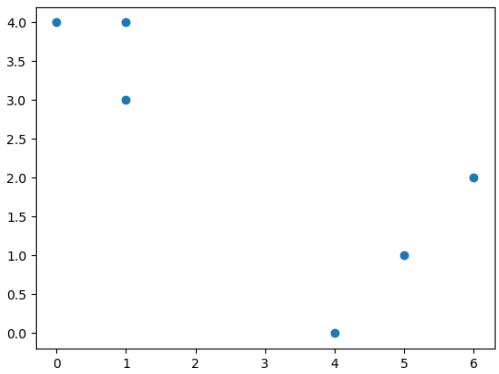
(e) It is mentioned in this chapter that at each fusion in the dendrogram, the position of .the two clusters being fused can be swapped without changing the meaning of the dendrogram. Draw a dendrogram that is equivalent to the dendrogram in (a), for which two or more of the leaves are repositioned, but for which the meaning of the dendrogram is the same.



2. Ch12-Prob3: In this problem, you will perform K-means clustering manually, with K = 2, on a small example with n = 6 observations and p = 2 features. The observations are as follows.

Obs.	X_1	X_2
1	1	4
2	1	3
3	0	4
4	5	1
5	6	2
6	4	0

(a) Plot the observations.



(b) Randomly assign a cluster label to each observation. You can use the np.random.choice() function to do this. Report the cluster labels for each observation.

```
np.random.seed(77)
X[:,2] = np.random.choice(2,6)
X[:,2]
```

```
array([1, 1, 0, 0, 1, 1])
```

(c) Compute the centroid for each cluster.

```
C = np.zeros((2,2))
for i in range(6):
  C[0,:] = np.mean(X[X[:,2]==0][:,:2],axis=0)
  C[1,:] = np.mean(X[X[:,2]==1][:,:2],axis=0)
C
```

```
array([[2.5 , 2.5 ],
       [3.  , 2.25]])
```

C1=(2.5, 2.5), C2=(3, 2.25)

(d) Assign each observation to the centroid to which it is closest, in terms of Euclidean distance. Report the cluster labels for each observation.

```
for i in range(6):
  if np.sum(np.square(X[i,:2]-C[0,:])) <= np.sum(np.square(X[i,:2]-C[1,:])): # if C1 is closer to the sample
    X[i,2] = 0 # assign cluster 0
  else:
    X[i,2] = 1
X[:,2]
```

```
array([0, 0, 0, 1, 1, 1])
```

The first three samples are in the cluster 0 and the others are in the cluster 1.

(e) Repeat (c) and (d) until the answers obtained stop changing.

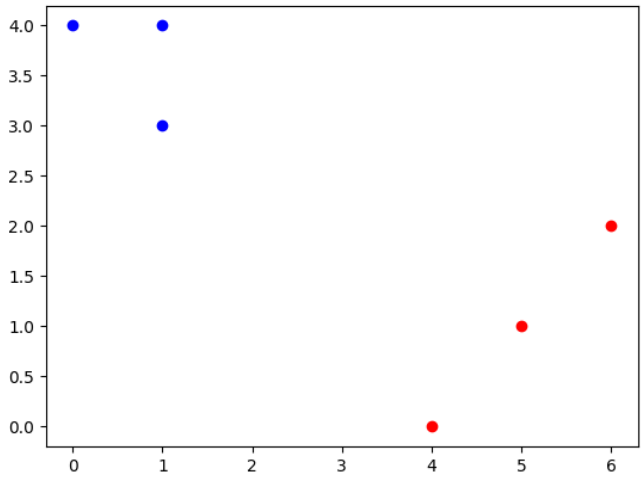
```
cluster_after = np.empty(6)
while (cluster_after != cluster_before).any():
  for i in range(6):
    C[0,:] = np.mean(X[X[:,2]==0][:,:2],axis=0)
    C[1,:] = np.mean(X[X[:,2]==1][:,:2],axis=0)
  for i in range(6):
    if np.sum(np.square(X[i,:2]-C[0,:])) <= np.sum(np.square(X[i,:2]-C[1,:])): # if C1 is closer to the sample
      X[i,2] = 0 # assign cluster 0
    else:
      X[i,2] = 1
  cluster_after = X[:,2]
print(cluster_before)
print(cluster_after)
```

```
[0 0 0 1 1 1]
[0. 0. 0. 1. 1. 1.]
```

The first three samples are in the cluster 0 and the others are in the cluster 1.

The result did not change from (d).

(f) In your plot from (a), color the observations according to the cluster labels obtained.



Blue dots: cluster 0 / Red dots: cluster 1

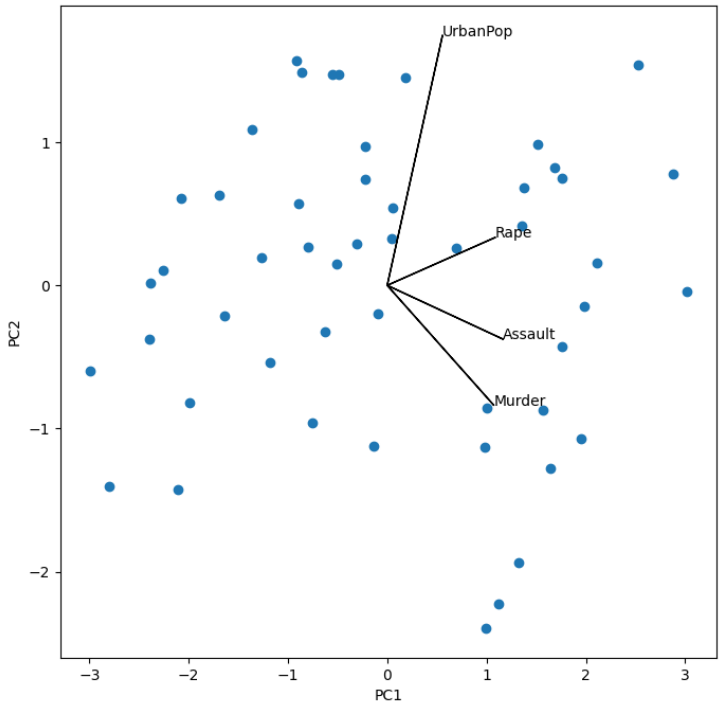
3. Ch12-Prob8: In Section 12.2.3, a formula for calculating PVE was given in Equation 12.10. We also saw that the PVE can be obtained using the explained_variance_ratio_ attribute of a fitted PCA() estimator. On the USArrests data, calculate PVE in two ways:

(a) Using the explained_variance_ratio_ output of the fitted PCA() estimator, as was done in Section 12.2.3.

```
pcaUS.explained_variance_ratio_
```

```
array([0.62006039, 0.24744129, 0.0891408 , 0.04335752])
```

The biplot is as follows.



(b) By applying Equation 12.10 directly. The loadings are stored as the components_ attribute of the fitted PCA() estimator. Use those loadings in Equation 12.10 to obtain the PVE. These two approaches should give the same results.

Hint: You will only obtain the same results in (a) and (b) if the same data is used in both cases. For instance, if in (a) you performed PCA() using centered and scaled variables, then you must center and scale the variables before applying Equation 12.10 in (b).

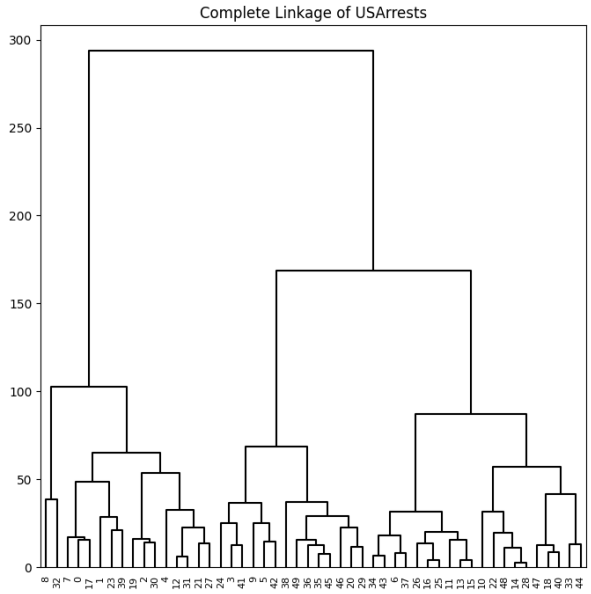
```
pc = []
for p in range(4):
  num = 0
  din = 0
  for i in range(n):
    temp = 0
    for j in range(4):
      temp += phi[p][j]*USArrests_scaled[i][j]
      din += USArrests_scaled[i][j]**2
    num += temp**2
  pc.append(num/din)
pc
```

```
[0.6200603947873733,
 0.24744128813496027,
 0.08914079514520752,
 0.043357521932458835]
```

The result was the same as (a).

4. Ch12-Prob9: Consider the USArrests data. We will now perform hierarchical clustering on the states.

   (a) Using hierarchical clustering with complete linkage and Euclidean distance, cluster the states.

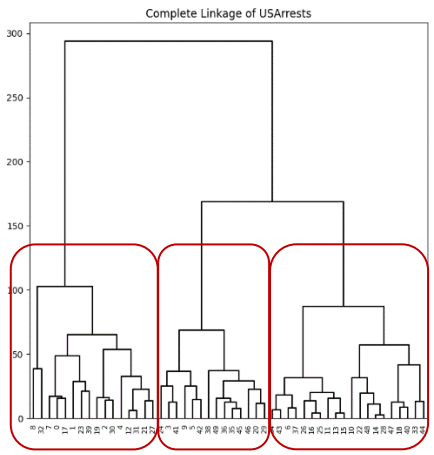


   (b) Cut the dendrogram at a height that results in three distinct clusters. Which states belong to which clusters?

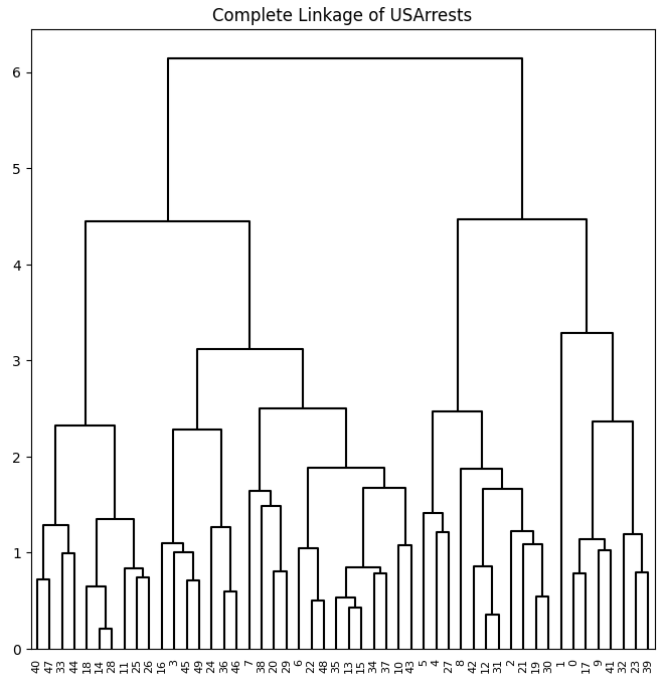
```
print(cut_tree(linkage_comp, n_clusters=3).T[0][:10])
print(cut_tree(linkage_comp, n_clusters=3).T[0][10:20])
print(cut_tree(linkage_comp, n_clusters=3).T[0][20:30])
print(cut_tree(linkage_comp, n_clusters=3).T[0][30:40])
print(cut_tree(linkage_comp, n_clusters=3).T[0][40:50])
```

```
[0 0 0 1 0 1 2 0 0 1]
[2 2 0 2 2 2 2 0 2 0]
[1 0 2 0 1 2 2 0 2 1]
[0 0 0 2 2 1 1 2 1 0]
[2 1 1 2 2 1 1 2 2 1]
```

   We can distinguish from the dendrogram above like below.



(c) Hierarchically cluster the states using complete linkage and Euclidean distance, after scaling the variables to have standard deviation one.



(d) What effect does scaling the variables have on the hierarchical clustering obtained? In your opinion, should the variables be scaled before the inter-observation dissimilarities are computed? Provide a justification for your answer.

Of course, the dissimilarities became smaller. On top of that, the structure of the clustering is completely changed. Before scaling, there were three big clusters with dissimilarity 1/3 of the maximum value(1/3 * 300 = 100). However, after scaling, there are about five big clusters with dissimilarity 1/3 of the maximum value(1/3 * 6 = 2).

Therefore, the variables must not be scaled before the inter-observation dissimilarities are computed.

5. Ch12-Prob13: On the book website, www.statlearning.com, there is a gene expression data set (Ch12Ex13.csv) that consists of 40 tissue samples with measurements on 1,000 genes. The first 20 samples are from healthy patients, while the second 20 are from a diseased group.

(a) Load in the data using pd.read_csv(). You will need to select header = None.

The genetic data is shown below.

```
Genes = pd.read_csv("/content/Ch12Ex13.csv",header=None)
Genes.head()
```

	0	1	2	3	4	5	6	7	8
0	-0.961933	0.441803	-0.975005	1.417504	0.818815	0.316294	-0.024967	-0.063966	0.031497
1	-0.292526	-1.139267	0.195837	-1.281121	-0.251439	2.511997	-0.922206	0.059543	-1.409645
2	0.258788	-0.972845	0.588486	-0.800258	-1.820398	-2.058924	-0.064764	1.592124	-0.173117
3	-1.152132	-2.213168	-0.861525	0.630925	0.951772	-1.165724	-0.391559	1.063619	-0.350009
4	0.195783	0.593306	0.282992	0.247147	1.978668	-0.871018	-0.989715	-1.032253	-1.109654

5 rows × 40 columns

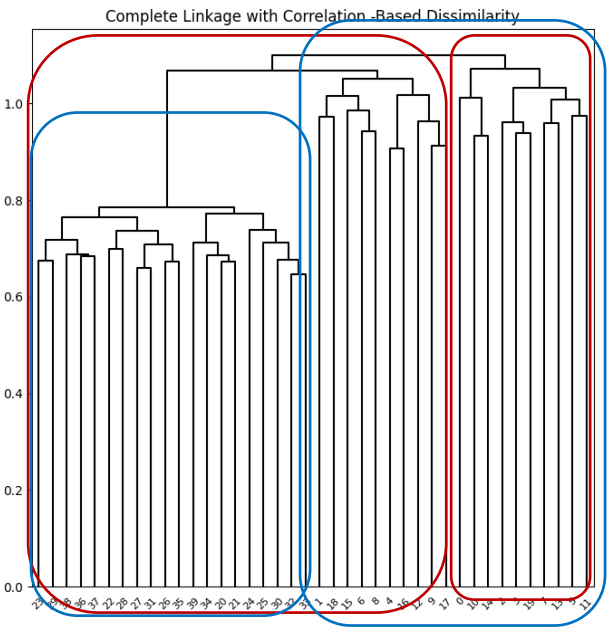
(b) Apply hierarchical clustering to the samples using correlation based distance, and plot the dendrogram. Do the genes separate the samples into the two groups? Do your results depend on the type of linkage used?

First, we get the dissimilarity matrix with the size 40 by 40 from the correlation matrix.

```
corD = 1 - np.corrcoef(Genes.transpose())
corD.shape
```

```
(40, 40)
```

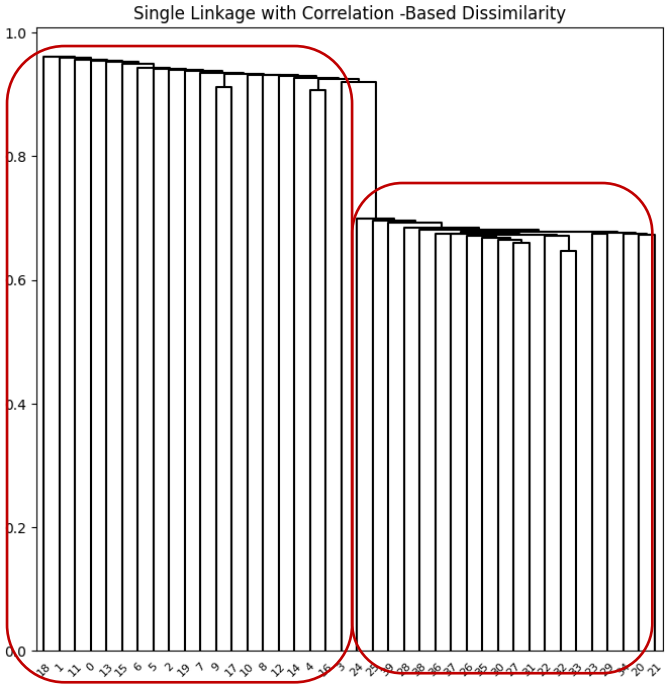
With complete linkage:



We can separate these into to like the red boxes.

But it is more intuitive to split into two with blue boxes.

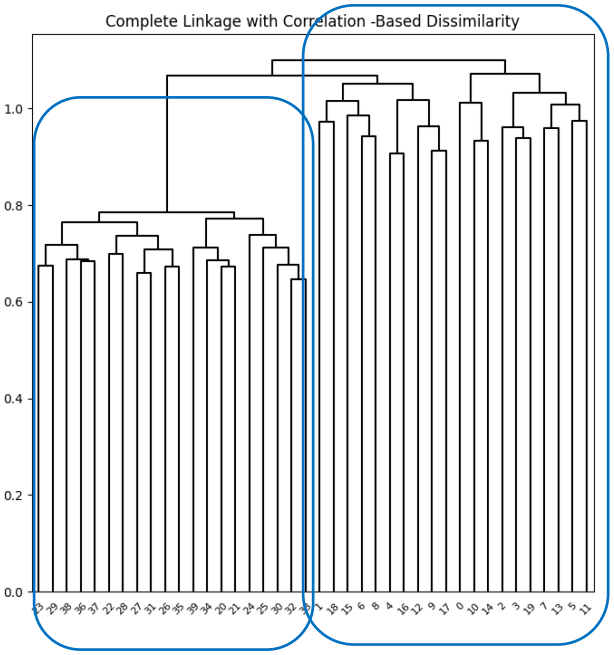
With single linkage:



We can see that there are two big clusters with different height.

Essentially, results does not depend on the type of linkage used.

(c) Your collaborator wants to know which genes differ the most across the two groups. Suggest a way to answer this question, and apply it here.



By looking at this clustering, I split these into two with the blue boxes.

```
c1 = [23,29,38,36,37,22,28,27,31,26,35,39,34,20,21,24,25,30,32,33]
c2 = [1,18,15,6,8,4,16,12,9,17,0,10,14,2,3,19,7,13,5,11]
```

```
l1 = len(c1)
l2 = len(c2)
avg1 = np.zeros(1000)
avg2 = np.zeros(1000)
for i in range(l1):
  avg1 += Genes.iloc[:,c1[i]]
avg1 = avg1/l1
for i in range(l2):
  avg2 += Genes.iloc[:,c2[i]]
avg2 = avg1/l2

diff = np.abs(avg1-avg2)
imp_gene = np.argmax(diff)
print(imp_gene)
```

```
550
```

두 그룹 간에 Gene data 값을 평균을 낸 이후에, 그 평균 값의 차이가 가장 큰 gene의 index를 찾았습니다. 그렇게 나온 gene이 551번째 gene이고, 저는 551번째 gene이 가장 핵심적인 요소라고 판단하였습니다.

실제로, 두 cluster간에 551번째 gene의 값을 평균을 내보면 다음과 같습니다.

```
print(avg1[550],avg2[550])
```

```
2.5246711 0.126233555
```

2.52 vs 0.12 로, 꽤 많은 차이를 보임을 알 수 있습니다.

또한 원래 데이터를 살펴보아도 아프지 않은 사람은 아래와 같이 음수이거나 작은 값을,

-0.25011 -1.55321 0.839214 -0.16594 1.331579 -0.26848 -0.07793 -1.55705 2.260733 0.117051 1.693249 0.271343 0.500236 0.413331 0.00144 -0.1607 0.566021 0.634264 0.382365 -1.34087

아픈 사람은 아래와 같이 큰 값을 갖는 것을 볼 수 있습니다.

2.205769 1.241147 3.542545 1.88687 2.538947 2.265187 2.590185 3.395062 3.793863 1.86649 2.512442 2.084097 3.414882 1.71977 1.446271 4.18091 1.175673 2.442113 3.549796 2.641403

6. Ch13-Prob1: Suppose we test m null hypotheses, all of which are true. We control the Type I error for each null hypothesis at level α. For each subproblem, justify your answer.

   (a) In total, how many Type I errors do we expect to make?

   Since we control the Type I error rate at α for each individual test and we are conducting m tests, the $expected\ number\ of\ Type\ I\ errors = m \times \alpha$.

   (b) Suppose that the m tests that we perform are independent. What is the family-wise error rate associated with these m tests?

Hint: If two events A and B are independent, then $Pr(A \cap B) = Pr(A)\,Pr(B)$.

$$FWER(\alpha) = 1 - P(V = 0) = 1 - P(no\ type\ I\ error) = 1 - \prod_{i=1}^{m}(1-\alpha) = 1 - (1-\alpha)^m$$

(c) Suppose that m = 2, and that the p-values for the two tests are positively correlated, so that if one is small then the other will tend to be small as well, and if one is large then the other will tend to be large. How does the family-wise error rate associated with these m = 2 tests qualitatively compare to the answer in (b) with m = 2?
Hint: First, suppose that the two p-values are perfectly correlated.

Let's suppose α = 0.05. Then, if m tests are independent,

$$FWER(\alpha) = 1 - 0.95^2 = 0.0975 = 2\alpha - \alpha^2$$

However, if they are perfectly correlated,

$$FWER(\alpha) = P(either\ H_{01}\ or\ H_{02}\ is\ rejected) = P(H_{01}\ is\ rejected) = \alpha$$

Let the correlation coefficient $0 < \rho < 1$.

$$FWER(\alpha) = P(either\ H_{01}\ or\ H_{02}\ is\ rejected)$$

$$= P(H_{01}\ is\ rejected) + P(H_{02}\ is\ rejected) - P(both\ H_{01}\ and\ H_{02}\ are\ rejected)$$

$$\cong \alpha + \alpha - \rho\alpha = 2\alpha - \rho\alpha$$

Therefore, generally, the FWER is smaller if the p values are positively correlated and the correlation is significant enough.

(d) Suppose again that m = 2, but that now the p-values for the two tests are negatively correlated, so that if one is large then the other will tend to be small. How does the family-wise error rate associated with these m = 2 tests qualitatively compare to the answer in (b) with m = 2?
Hint: First, suppose that whenever one p-value is less than α, then the other will be greater than α. In other words, we can never reject both null hypotheses.

Let's suppose α = 0.05. Then, if m tests are independent,

$$FWER(\alpha) = 1 - 0.95^2 = 0.0975 = 2\alpha - \alpha^2$$

However, if they are perfectly correlated,

$$FWER(\alpha) = P(H_{01}\ is\ rejected) + P(H_{02}\ is\ rejected) - P(both\ H_{01}\ and\ H_{02}\ are\ rejected)$$

$$= P(H_{01}\ is\ rejected) + P(H_{02}\ is\ rejected) = 2\alpha$$

Let the correlation coefficient $-1 < \rho < 0$.

$$FWER(\alpha) = P(either\ H_{01}\ or\ H_{02}\ is\ rejected)$$

$$= P(H_{01}\ is\ rejected) + P(H_{02}\ is\ rejected) - P(both\ H_{01}\ and\ H_{02}\ are\ rejected)$$

$$\cong \alpha + \alpha - \rho\alpha = 2\alpha - \rho\alpha > 2\alpha$$

Therefore, generally, the FWER is larger if the p values are negatively correlated and the correlation is significant enough.

7. Ch13-Prob2: Suppose that we test m hypotheses, and control the Type I error for each hypothesis at level α. Assume that all m p-values are independent, and that all null hypotheses are true.

   (a) Let the random variable A_j equal 1 if the jth null hypothesis is rejected, and 0 otherwise. What is the distribution of A_j?

   $$A_j \sim Bern(\alpha), \qquad P(A_j = x) = \begin{cases} 1-\alpha, & if\ x = 0 \\ \alpha & if\ x = 1 \end{cases}$$

   (b) What is the distribution of $\sum_{j=1}^{m} A_j$?

   $$\sum_{j=1}^{m} A_j \sim Binomial(m, \alpha), \qquad P\left(\sum_{j=1}^{m} A_j = x\right) = \binom{m}{x} \alpha^x (1-\alpha)^{m-x}, 0 \le x \le m$$

   (c) What is the standard deviation of the number of Type I errors that we will make?

   $$Var\left(\sum_{j=1}^{m} A_j\right) = m\alpha(1-\alpha), \qquad \sigma = \sqrt{m\alpha(1-\alpha)}$$

8. Ch13-Prob4: Suppose we test m = 10 hypotheses, and obtain the p-values shown in Table 13.4.

Null Hypothesis	p-value
H_{01}	0.0011
H_{02}	0.031
H_{03}	0.017
H_{04}	0.32
H_{05}	0.11
H_{06}	0.90
H_{07}	0.07
H_{08}	0.006
H_{09}	0.004
H_{10}	0.0009

TABLE 13.4. *p-values for Exercise 4.*

   (a) Suppose that we wish to control the Type I error for each null hypothesis at level α = 0.05. Which null hypotheses will we reject?

   We reject $H_{01}, H_{02}, H_{03}, H_{08}, H_{09}\ and\ H_{10}$

   (b) Now suppose that we wish to control the FWER at level α = 0.05. Which null

hypotheses will we reject? Justify your answer.

We can apply the Bonferroni method.

$$\alpha_{Bonf} = \frac{\alpha}{Number\ of\ Tests} = \frac{0.05}{10} = 0.005$$

We reject a null hypothesis if the p-value is smaller than 0.005

Therefore, We reject $H_{01}, H_{09}\ and\ H_{10}$.

(c) Now suppose that we wish to control the FDR at level q = 0.05. Which null hypotheses will we reject? Justify your answer.

We can use the Benjamini-Hochberg procedure.

1. Order the p-values: 0.0009,0.0011,0.004,0.006,0.017,0.031,0.07,0.11,0.32,0.90.
2. Calculate the Benjamini-Hochberg critical values: $CV_i = \frac{i}{10} * 0.05$
3. Compare p-values to critical values:

   0.0009≤0.005 (Reject)

   0.0011≤0.01 (Reject)

   0.004≤0.015 (Reject)

   0.006≤0.02 (Reject)

   0.017≤0.025 (Reject)

   0.031>0.03 (Do not reject)

   0.07>0.035 (Do not reject)

   0.11>0.04 (Do not reject)

   0.32>0.045 (Do not reject)

   0.90>0.05 (Do not reject)

   Therefore, we reject $H_{10}, H_{01}, H_{09}, H_{08}\ and\ H_{03}$.

(d) Now suppose that we wish to control the FDR at level q = 0.2. Which null hypotheses will we reject? Justify your answer.

We can use the Benjamini-Hochberg procedure.

1. Order the p-values: 0.0009,0.0011,0.004,0.006,0.017,0.031,0.07,0.11,0.32,0.90.

2. Calculate the Benjamini-Hochberg critical values: $CV_i = \frac{i}{10} * 0.2$

3. Compare p-values to critical values:

   0.0009≤0.02 (Reject)

   0.0011≤0.04 (Reject)

   0.004≤0.06 (Reject)

   0.006≤0.08 (Reject)

   0.017≤0.10 (Reject)

   0.031≤0.12 (Reject)

   0.07≤0.14 (Reject)

   0.11≤0.16 (Reject)

   0.32>0.18 (Do not reject)

   0.90>0.20 (Do not reject)

   Therefore, we reject $H_{10}, H_{01}, H_{09}, H_{08}, H_{03}, H_{02}, H_{07}\ and\ H_{05}$.

(e) Of the null hypotheses rejected at FDR level q = 0.2, approximately how many are false positives? Justify your answer.

$$FDR = \frac{Number\ of\ False\ Positives}{Number\ of\ Rejected\ Hypotheses}$$

$$Number\ of\ False\ Positives \cong FDR * Number\ of\ Rejected\ Hypotheses = 0.2 * 8 = 1.6$$

9. Ch13-Prob6: 6. For each of the three panels in Figure 13.3, answer the following questions:

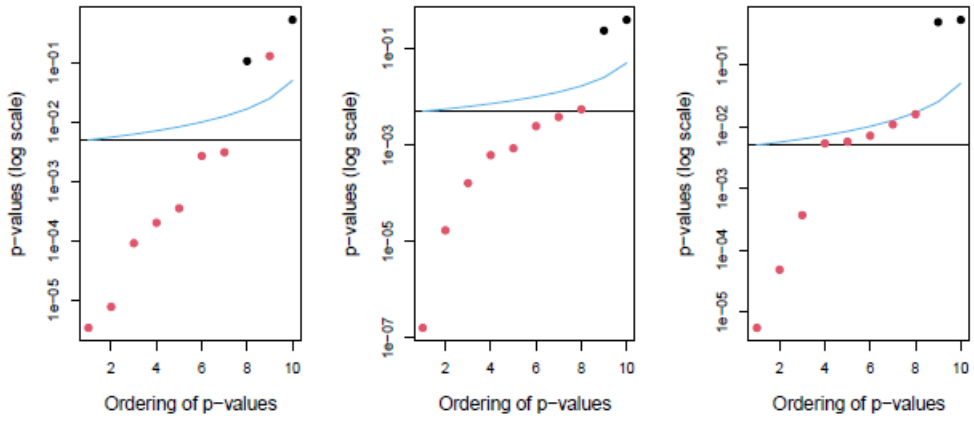


FIGURE 13.3. *Each panel displays, for a separate simulation, the sorted p-values for tests of* $m = 10$ *null hypotheses. The p-values corresponding to the* $m_0 = 2$ *true null hypotheses are displayed in black, and the rest are in red. When controlling the FWER at level* 0.05*, the Bonferroni procedure rejects all null hypotheses that fall below the black line, and the Holm procedure rejects all null hypotheses that fall below the blue line. The region between the blue and black lines indicates null hypotheses that are rejected using the Holm procedure but not using the Bonferroni procedure. In the center panel, the Holm procedure rejects one more null hypothesis than the Bonferroni procedure. In the right-hand panel, it rejects five more null hypotheses.*

(a) How many false positives, false negatives, true positives, true negatives, Type I errors, and Type II errors result from applying the Bonferroni procedure to control the FWER at level α = 0.05?

Panel	FP	FN	TP	TN	Type I	Type II
Left	0	1	2	7	0	1
Middle	0	1	2	7	0	1
Right	0	5	2	3	0	5

(b) How many false positives, false negatives, true positives, true negatives, Type I errors, and Type II errors result from applying the Holm procedure to control the FWER at level α = 0.05?

Panel	FP	FN	TP	TN	Type I	Type II
Left	0	1	2	7	0	1
Middle	0	0	2	8	0	0
Right	0	0	2	8	0	0

(c) What is the false discovery proportion associated with using the Bonferroni procedure to control the FWER at level α = 0.05?

FDR = $\frac{\#\ of\ FP}{\max(\#\ of\ rejected\ Hypotheses, 1)}$

Panel	FDR
Left	0
Middle	0
Right	0

(d) What is the false discovery proportion associated with using the Holm procedure to control the FWER at level α = 0.05?

Panel	FDR
Left	0
Middle	0
Right	0

(e) How would the answers to (a) and (c) change if we instead used the Bonferroni procedure to control the FWER at level α = 0.001?

a)

Panel	FP	FN	TP	TN	Type I	Type II
Left	0	6	2	2	0	6
Middle	0	6	2	2	0	6
Right	0	6	2	2	0	6

c)

Panel	FDR
Left	0
Middle	0
Right	0

10. Ch13-Prob8: In this problem, we will simulate data from m = 100 fund managers.

```
rng = np.random.default_rng(1)
n, m = 20, 100
X = rng.normal(size=(n, m))
```

These data represent each fund manager's percentage returns for each of n = 20 months. We wish to test the null hypothesis that each fund manager's percentage returns have population mean equal to zero. Notice that we simulated the data in such a way that each fund manager's

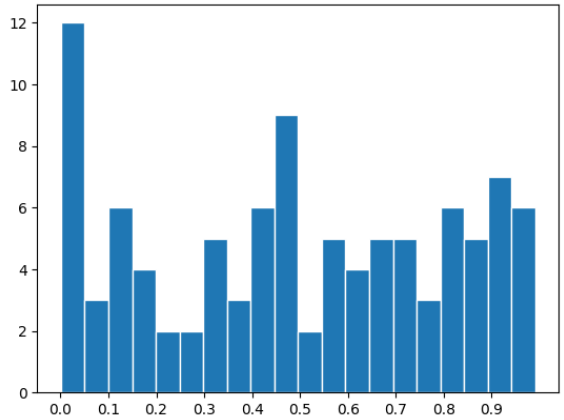
percentage returns do have population mean zero; in other words, all m null hypotheses are true.

(a) Conduct a one-sample t-test for each fund manager, and plot a histogram of the p-values obtained.

I changed the random seed to 77.

```
rng = np.random.default_rng(77)
n, m = 20, 100
X = rng.normal(size=(n, m))
```

The histogram of the p-values is as follows.



(b) If we control Type I error for each null hypothesis at level α = 0.05, then how many null hypotheses do we reject?

```
pd.crosstab(decision, truth, rownames=['Decision'], colnames=['H0'])
```

Decision	H0 True
Reject H0	12
Do not reject H0	88

We reject 12 null hypotheses.

(c) If we control the FWER at level 0.05, then how many null hypotheses do we reject?

```
[76] reject, bonf = mult_test(p_values, method = "bonferroni")[:2]
     num_rej_bonf = np.sum(reject*np.ones(100))
     print("number of rejection with Bonferroni method =", num_rej_bonf)

     number of rejection with Bonferroni method = 0.0
```

```
[77] reject, holm = mult_test(p_values, method = "holm", alpha=0.05)[:2]
     num_rej_holm = np.sum(reject*np.ones(100))
     print("number of rejection with Holm's method =", num_rej_holm)

     number of rejection with Holm's method = 0.0
```

For both Bonferroni method and Holm's method, we reject no null hypothesis.

(d) If we control the FDR at level 0.05, then how many null hypotheses do we reject?

```
(q_values <= 0.05).sum()
```

```
0
```

We do not reject any null hypothesis.

(e) Now suppose we "cherry-pick" the 10 fund managers who perform the best in our data. If we control the FWER for just these 10 fund managers at level 0.05, then how many null hypotheses do we reject? If we control the FDR for just these 10 fund managers at level 0.05, then how many null hypotheses do we reject?

I chose 10 managers who had the top 10 percentage return in total.

Their p-values was like below.

```
best_10_idx = np.argsort(performance)[-10:]
best_10_p_values = [p_values[i] for i in best_10_idx]
best_10_p_values
```

```
[0.03822965999073228,
 0.008889217554356694,
 0.014875325358609014,
 0.03659608132400176,
 0.016122165219132574,
 0.049361809583004526,
 0.05989065019398282,
 0.003099880255080984,
 0.0010281118533028986,
 0.0011726756527197465]
```

For both Bonferroni method and Holm's method, we reject 3 null hypotheses.

```
reject, bonf = mult_test(best_10_p_values, method = "bonferroni")[:2]
num_rej_bonf = np.sum(reject*np.ones(10))
print("number of rejection with Bonferroni method =", num_rej_bonf)
```

```
number of rejection with Bonferroni method = 3.0
```

```
reject, holm = mult_test(best_10_p_values, method = "holm", alpha=0.05)[:2]
num_rej_holm = np.sum(reject*np.ones(10))
print("number of rejection with Holm's method =", num_rej_holm)
```

```
number of rejection with Holm's method = 3.0
```

(f) Explain why the analysis in (e) is misleading.

Hint: The standard approaches for controlling the FWER and FDR assume that all tested null hypotheses are adjusted for multiplicity, and that no "cherry-picking" of the smallest p-values has occurred. What goes wrong if we cherry-pick?

Since we chose the top 10 managers who showed the best performances, their p-values tend to be smaller than the average of the population. Moreover, since m has become smaller from 100 to 10, the cutoff has been loose for rejecting a null hypothesis. Therefore it rejects more percentage of null hypotheses than it does for the population. (From 0% it became 30%)