Assignment 7 By Team 1

Odianosen Akhibi, Shun-Lung Chang, Juliana Nair

This study was conducted in R, and the source code can be found here.

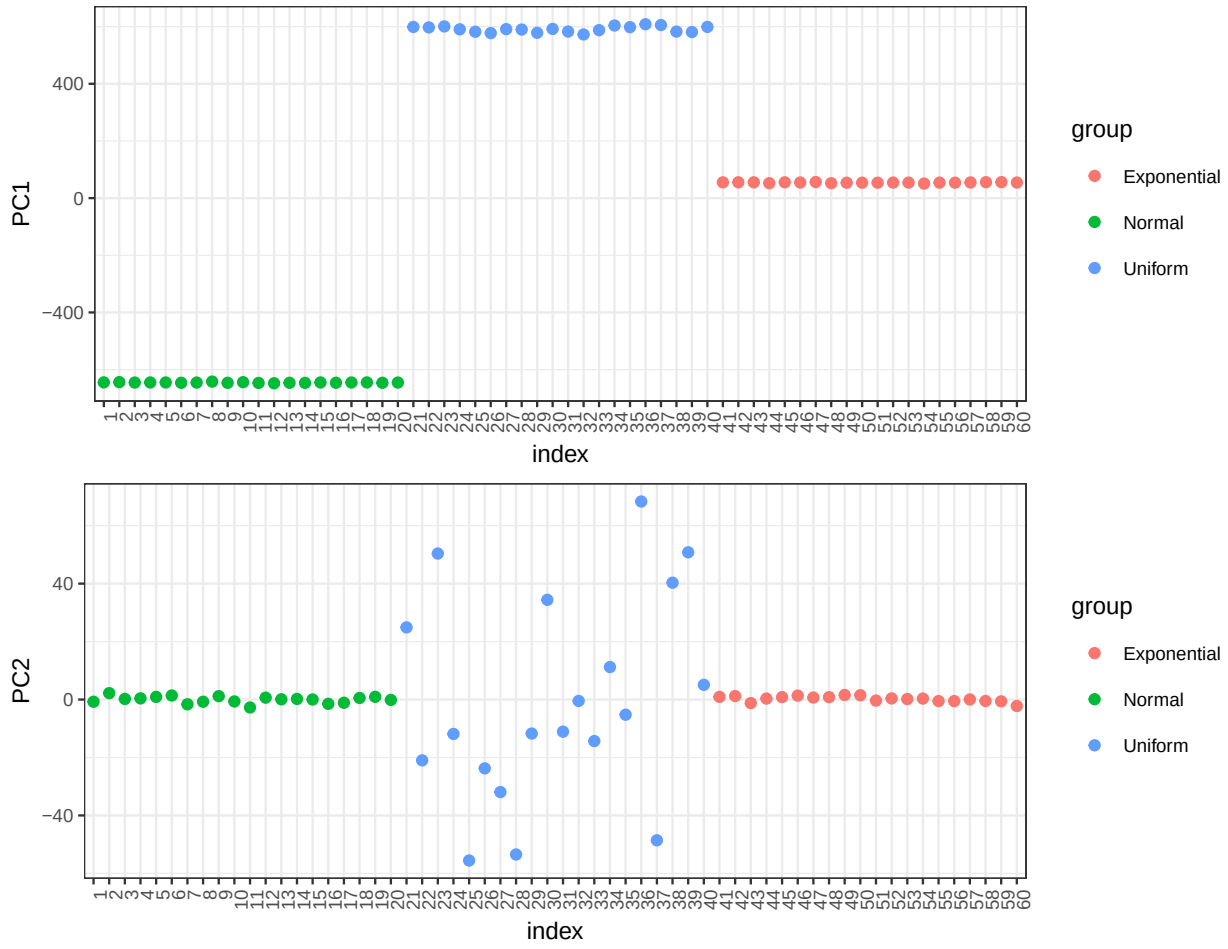
1. Generate a simulated data set with 20 observations in each of three classes (i.e. 60 observations total), and 50 variables.

```
# set the random number seed
set.seed(45)

normal <- rnorm(1000, mean = 100, sd = 1) %>%
    matrix(20, 50) %>%
    data.frame()
uniform <- runif(1000, min = -100, max = -50) %>%
    matrix(20, 50) %>%
    data.frame()
exponential <- rexp(1000, rate = 1) %>%
    matrix(20, 50) %>%
    data.frame()

dat <- rbind(normal, uniform, exponential)

dim(dat)
```

```
[1] 60 50
```

2. Perform PCA on the 60 observations and plot the first two principal component score vectors. Use a different color to indicate the observations in each of the three classes.

```
pca <- prcomp(dat)
```



3. Perform K-means clustering of the observations with K = 3. How well do the clusters that you obtained in K-means clustering compare to the true class labels?

```
kc_3 <- kmeans(dat, center = 3)

table(kc_3$cluster, c(rep(1, 20), rep(2, 20), rep(3, 20))) %>% kable()
```

1	2	3
0	9	0
0	11	0
20	0	20

4. Perform K-means clustering with K = 2. Describe your results.

```
kc_2 <- kmeans(dat, center = 2)
```

```r
table(kc_2$cluster, c(rep(1, 20), rep(2, 20), rep(3, 20))) %>% kable()
```

1	2	3
20	0	20
0	20	0

5. Now perform K-means clustering with K = 4, and describe your results.

```r
kc_4 <- kmeans(dat, center = 4)

table(kc_4$cluster, c(rep(1, 20), rep(2, 20), rep(3, 20))) %>% kable()
```

1	2	3
9	0	0
5	0	0
6	0	0
0	20	20

6. Now perform K-means clustering with K = 3 on the first two principal component score vectors, rather than on the raw data. That is, pserform K-means clustering on the 60 × 2 matrix of which the first column is the first principal component score vector, and the second column is the second principal component score vector. Comment on the results.

```r
kc_pc_3 <- kmeans(data.frame(two_pcs$PC1, two_pcs$PC2), centers = 3)

table(kc_pc_3$cluster, c(rep(1, 20), rep(2, 20), rep(3, 20))) %>% kable()
```

1	2	3
0	7	0
20	0	20
0	13	0

7. Using the `scale()` function, perform K-means clustering with K = 3 on the data after scaling each variable to have standard deviation one. How do these results compare to those obtained in (3)? Explain.

```
dat_scaled <- scale(dat)
kc_3 <- kmeans(dat_scaled, center = 3)

table(kc_3$cluster, c(rep(1, 20), rep(2, 20), rep(3, 20))) %>% kable()
```

1	2	3
0	0	20
0	20	0
20	0	0

8. Use the scaled variables and run a PCA on them. Now perform K-means clustering with K = 3 on the first two principal component score vectors, rather than on the raw data. How do these results compare to those obtained in (3) and (7)? Explain.

```
pca_scaled <- prcomp(dat, center = TRUE, scale. = TRUE)

kc_pc_scaled_3 <- kmeans(data.frame(pca_scaled$x[, 1], pca_scaled$x[, 2]), centers = 3)

table(kc_pc_scaled_3$cluster, c(rep(1, 20), rep(2, 20), rep(3, 20))) %>% kable()
```

1	2	3
0	0	20
20	0	0
0	20	0