Homework 11

Eric Zou

Honor code:

"I have neither given nor received unauthorized assistance on this assignment." Type your initials here.

I receive help from " " and give help to " ".

Problem 1

In this problem, you will generate simulated data, and then perform PCA and K-means clustering on the data.

(a) Generate a simulated data set with 50 observations in each of three classes (i.e. 150 observations total), and 3 variables using the code below.

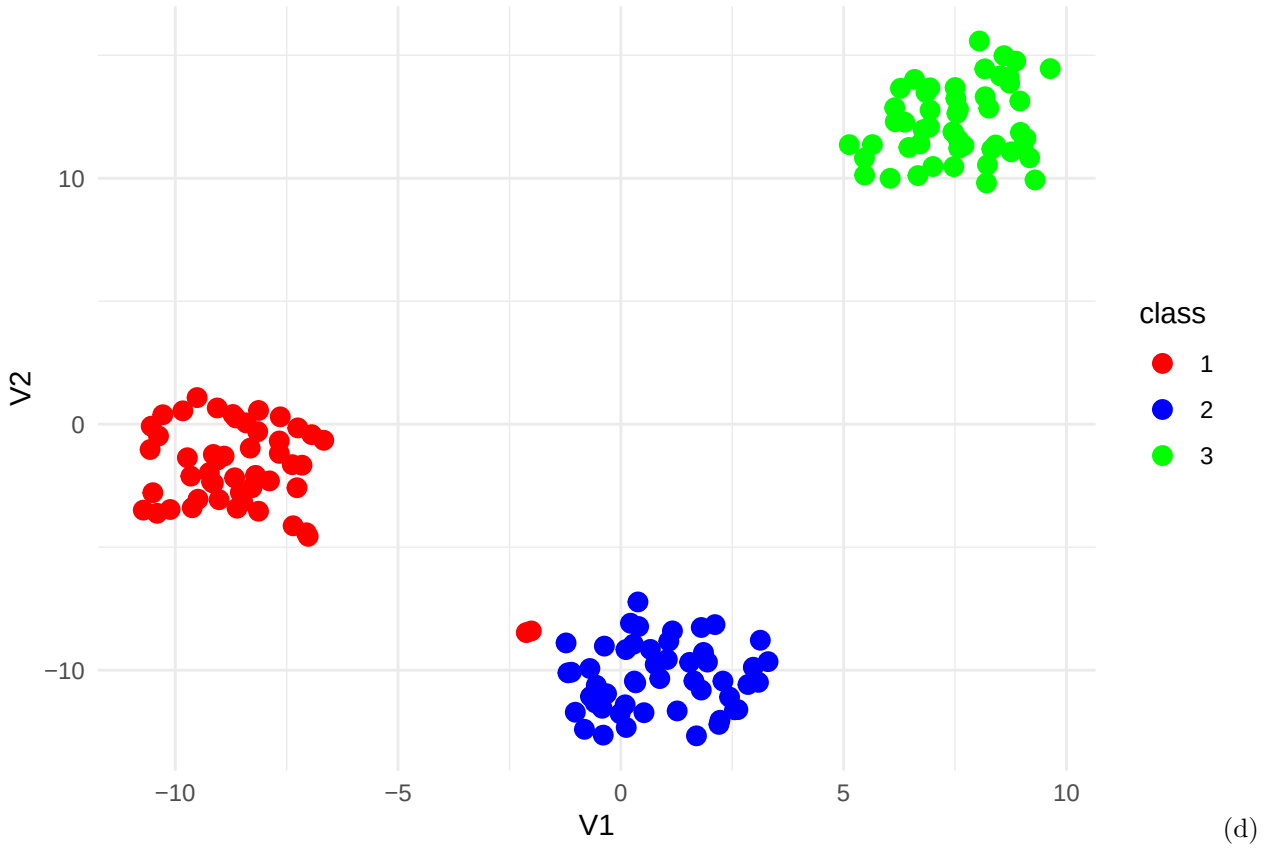
```
set.seed(111)
x=matrix(rnorm(150*3), ncol=3)
x[1:50,1]=x[1:50,1]+3
x[51:100,2]=x[1:50,2]-4
x[101:150, 3] = x[1:50,3]+5
```

(b) Perform PCA on the 150 observations and plot the first two principal component score vectors. Use a different colors to indicate the observations in each of the three classes (Hints: set groups in ggbiplot as the as.factor(rep(c(1,2,3),each=50))).

library(ggbiplot) library(ggplot2) pca_result = prcomp(x, scale = TRUE) ggbiplot(pca_result, obs.scale = 1, var.scale = 1, groups = as.factor(rep(c(1,2,3), each=50)), ellipse = TRUE, circle = TRUE) + scale_color_discrete(name = '') + theme_minimal()

(c) Perform tSNE on the 150 observations and plot the first two principal component score vectors. Use a different colors to indicate the observations in each of the three classes (Hints: map as.factor(rep(c(1,2,3),each=50)) to color when you draw plot using ggplot())

```
library(Rtsne)
library(ggplot2)
tsne_result = Rtsne(x, dims = 2, perplexity = 30, theta = 0.5, check_duplicates = FALSE)
tsne_data <- as.data.frame(tsne_result$Y)
tsne_data$class <- as.factor(rep(c(1, 2, 3), each = 50))
ggplot(tsne_data, aes(x = V1, y = V2, color = class)) +
  geom_point(size = 3) +
  scale_color_manual(values = c("red", "blue", "green")) +
  theme_minimal()
```



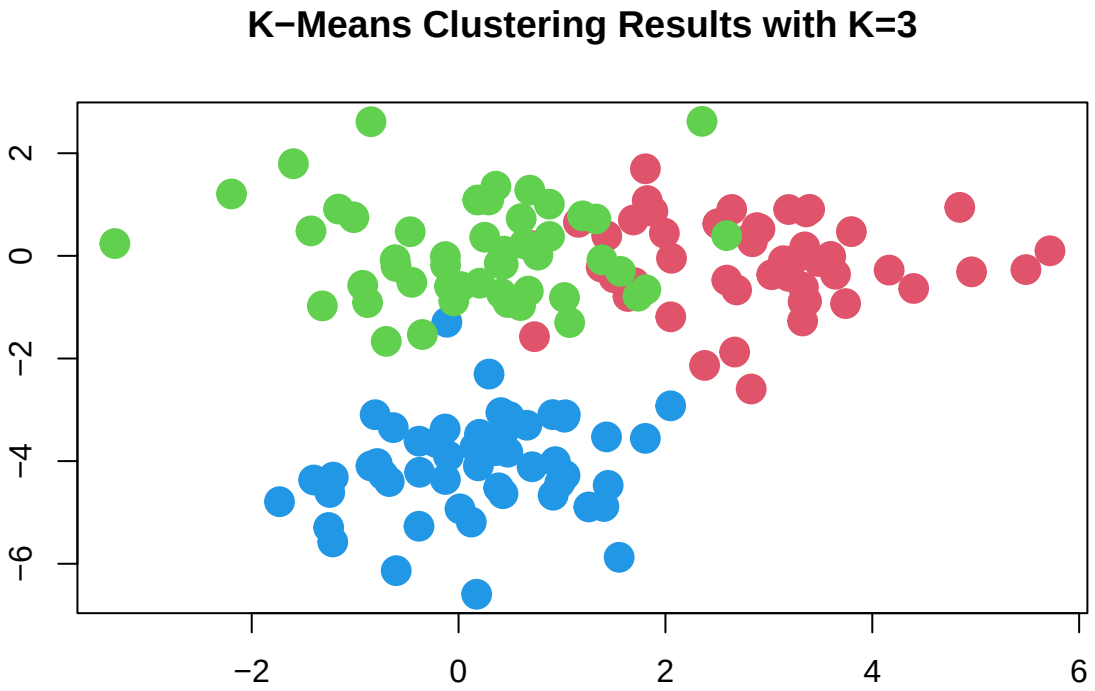
(d)

Calculate silhouette scores for a range of values of K (K=2, 3, . . . , 8) using the K-means Clustering algorithm. Visualize these scores with an appropriate plot to determine the optimal number of clusters.

```
km.out=kmeans(x,3,nstart=20)
km.out$cluster
```

```
##   [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 3 1 1 1 1 1 1
##  [38] 1 1 1 1 1 1 1 1 1 1 1 1 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
##  [75] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 2 2 2 2 2 2 2 2 2 2 2
## [112] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [149] 2 2
```

```
plot(x, col=(km.out$cluster+1), main="K-Means Clustering Results with K=3", xlab="", ylab="", pch=20, c
```



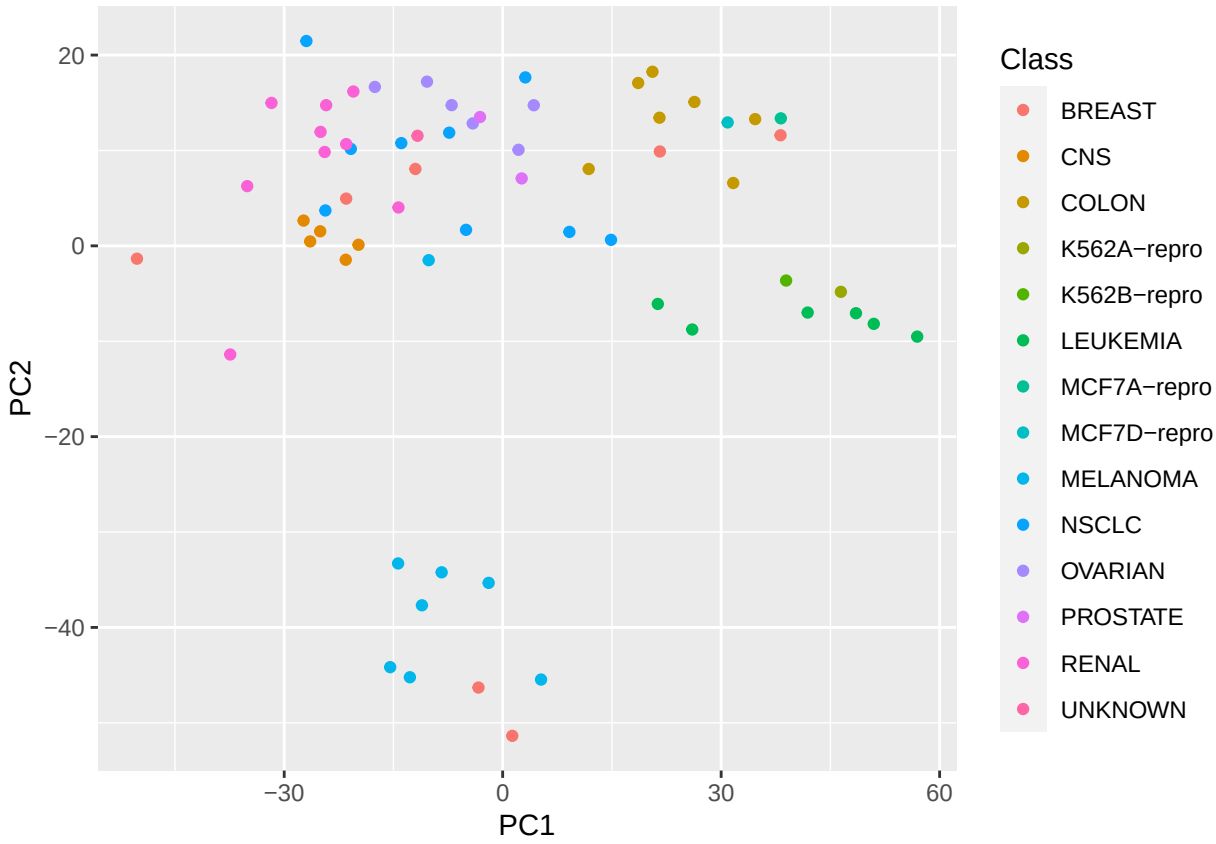
Problem 2

Load the NCI60 data in `ISLR` package which consists of 6,830 gene expression measurements on 64 cancer cell lines. The goal is finding out whether or not the observations cluster into distinct types of cancer.

```
library(ISLR)
dat = as.data.frame(NCI60$data)
```

(a) Perform PCA on the 64 observations and plot the first two principal component score vectors (Use dat.pca = prcomp(dat) for PCA analysis). Use a different colors to indicate the observations in each of the classes. (set groups as.factor(NCI60$labs))

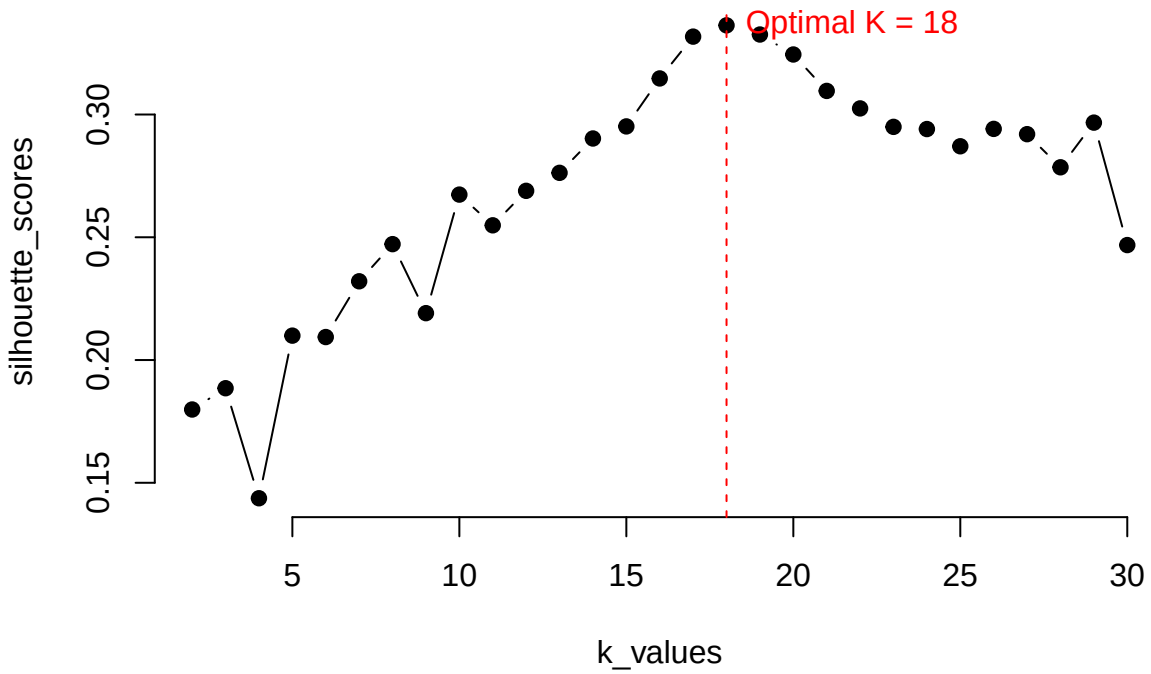
```
library(ggplot2)
dat.pca = prcomp(dat)
pca_data <- data.frame(PC1 = dat.pca$x[, 1], PC2 = dat.pca$x[, 2], Class = as.factor(NCI60$labs))
ggplot(pca_data, aes(x = PC1, y = PC2, color = Class)) +
  geom_point()
```



(b) Perform dimensionality reduction on the 'dat' project by projecting it onto the first 20 principal components (PCs) (Hint: x = dat.pca$x[,1:20] is the reduced-dimensional data). Utilize this reduced-dimensional data for subsequent clustering analysis. Calculate silhouette scores for a range of values of K (K=2, 3, . . . , 30) using the K-means Clustering algorithm. Visualize these scores with an appropriate plot to determine the optimal number of clusters.

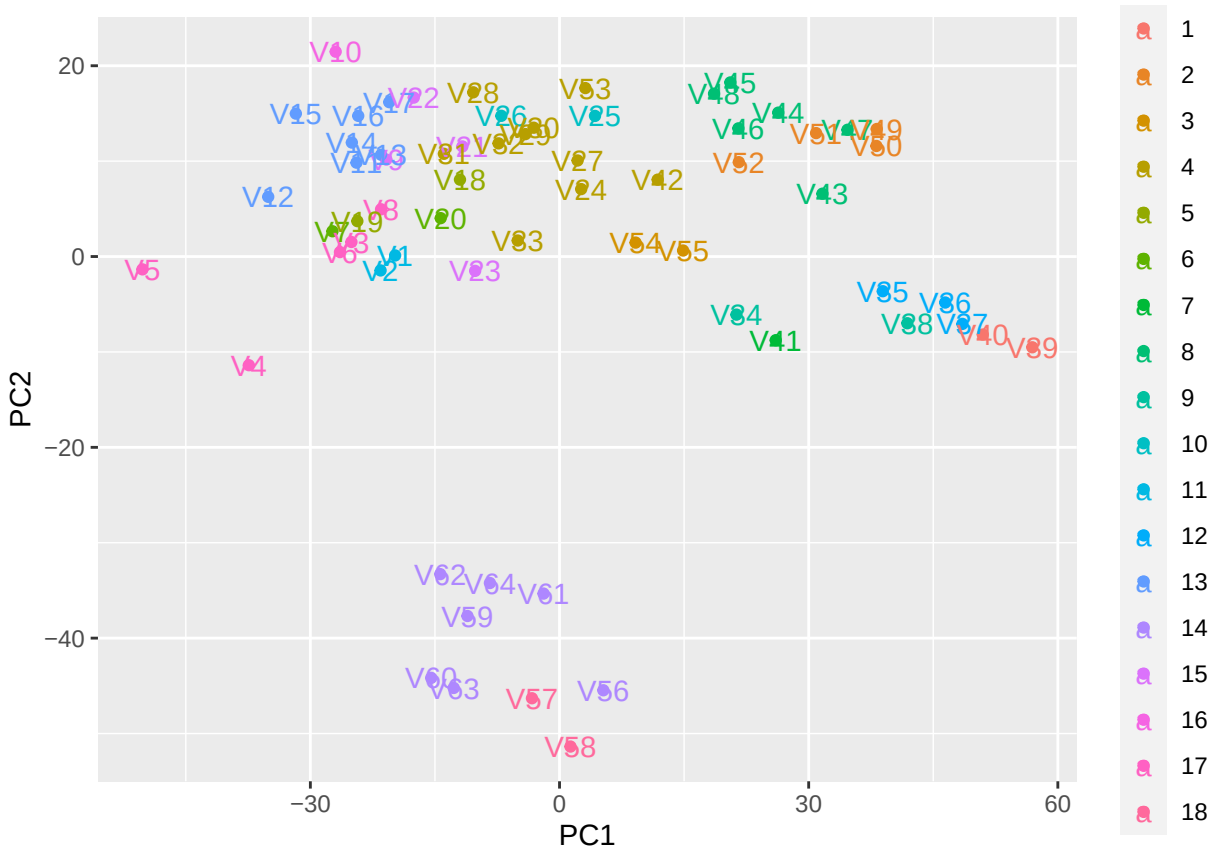
```
reduced_dat <- dat.pca$x[, 1:20]

library(cluster)
calculate_silhouette <- function(data, k) {
  kmeans_model <- kmeans(data, centers = k, nstart = 10)
  silhouette_score <- silhouette(kmeans_model$cluster, dist(data))
  return(mean(silhouette_score[, 3]))
}
k_values <- 2:30
silhouette_scores <- sapply(k_values, function(k) calculate_silhouette(reduced_dat, k))
plot(k_values, silhouette_scores, type = "b", pch = 19, frame = FALSE)
optimal_k <- k_values[which.max(silhouette_scores)]
abline(v = optimal_k, col = "red", lty = 2)
text(optimal_k, max(silhouette_scores), labels = paste("Optimal K =", optimal_k), pos = 4, col = "red")
```



(c) Create a biplot representing the scores on the first 2 principal components, and color-code the data points based on the optimal clustering labels obtained in (b).

```
optimal_kmeans <- kmeans(reduced_dat, centers = optimal_k, nstart = 10)
biplot_data <- data.frame(PC1 = dat.pca$x[, 1], PC2 = dat.pca$x[, 2], Cluster = as.factor(optimal_kmeans
ggplot(biplot_data, aes(x = PC1, y = PC2, color = Cluster)) +
  geom_point() +
  geom_text(aes(label = rownames(biplot_data)))
```


PC2
PC1
20
0
−20
−40
−30
0
30
60
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18