Assignment 10.2

Lincoln Brown

8/11/2021

Libraries

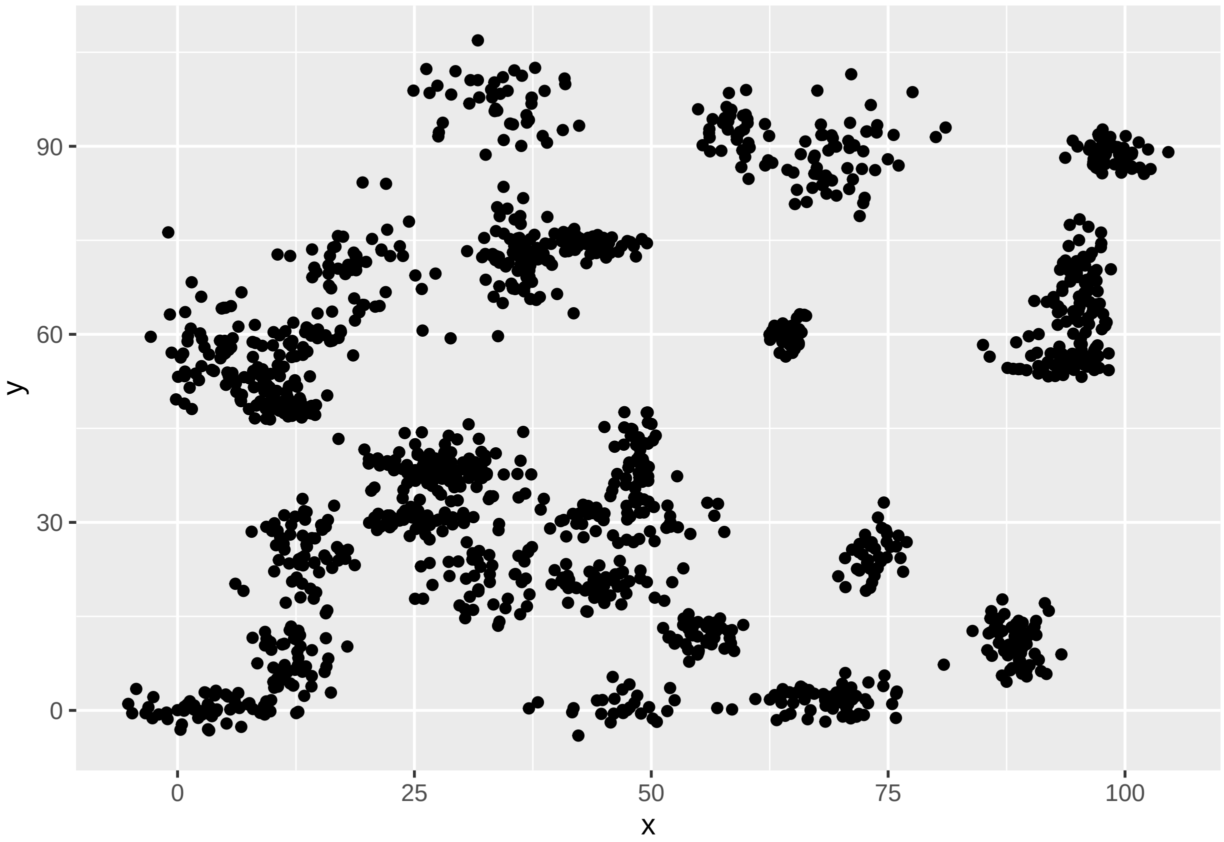
```
library(ggplot2)
library(class)
library(useful)
library(scales)
```

Binary Data

```
binary <- read.csv("/media/x/disk/School/DSC520/Datasets/binary-classifier-data.csv")
head(binary)
```

```
##   label        x        y
## 1     0 70.88469 83.17702
## 2     0 74.97176 87.92922
## 3     0 73.78333 92.20325
## 4     0 66.40747 81.10617
## 5     0 69.07399 84.53739
## 6     0 72.23616 86.38403
```

```
b_base <- ggplot(binary, aes(x=x, y=y))
b_base + geom_point()
```



Create sample data

```
#Total # of rows in data set
n_bin <- nrow(binary)

# Set 80% of the rows for training sample
n_bin_train <- round(0.80 * n_bin)

# Create a vector of indices which is an 80% random sample
set.seed(1)
bin_train_indices <- sample(1:n_bin, n_bin_train)

# Subset the data frame to training indices only
bin_train <- binary[bin_train_indices,]

#Exclude the training indices for test set
bin_test <- binary[-bin_train_indices,]
```

```
knn.3 <- knn(train=bin_train, test=bin_test, cl=bin_train$label, k=3)
cm3 <- table(bin_test$label, knn.3)
cm3
```

```
##    knn.3
##       0   1
##   0 149   4
##   1   4 143
```

```r
mc_err3 <- mean(knn.3 != bin_test$label)
acc03 <- (1-mc_err3)
cat("Accuracy with k=3 is:",percent(acc03))
```

```
## Accuracy with k=3 is: 97%
```

```r
knn.5 <- knn(train=bin_train, test=bin_test, cl=bin_train$label, k=5)
cm5 <- table(bin_test$label, knn.5)
cm5
```

```
##    knn.5
##       0   1
##   0 148   5
##   1   4 143
```

```r
mc_err5 <- mean(knn.5 != bin_test$label)
acc05 <- (1-mc_err5)
cat("Accuracy with k=5 is:",percent(acc05))
```

```
## Accuracy with k=5 is: 97%
```

```r
knn.10 <- knn(train=bin_train, test=bin_test, cl=bin_train$label, k=10)
cm10 <- table(bin_test$label, knn.10)
cm10
```

```
##    knn.10
##       0   1
##   0 146   7
##   1   3 144
```

```r
mc_err10 <- mean(knn.10 != bin_test$label)
acc10 <- (1-mc_err10)
cat("Accuracy with k=10 is:",percent(acc10))
```

```
## Accuracy with k=10 is: 97%
```

```r
knn.15 <- knn(train=bin_train, test=bin_test, cl=bin_train$label, k=15)
cm15 <- table(bin_test$label, knn.15)
cm15
```

```
##    knn.15
##       0   1
##   0 147   6
##   1   3 144
```

```r
mc_err15 <- mean(knn.15 != bin_test$label)
acc15 <- (1-mc_err15)
cat("Accuracy with k=15 is:",percent(acc15))
```

```
## Accuracy with k=15 is: 97%
```

```r
knn.20 <- knn(train=bin_train, test=bin_test, cl=bin_train$label, k=20)
cm20 <- table(bin_test$label, knn.20)
cm20
```

```
##    knn.20
##       0   1
##   0 147   6
##   1   2 145
```

```
mc_err20 <- mean(knn.20 != bin_test$label)
acc20 <- (1-mc_err20)
cat("Accuracy with k=20 is:",percent(acc20))
```

```
## Accuracy with k=20 is: 97%
```

```
knn.25 <- knn(train=bin_train, test=bin_test, cl=bin_train$label, k=25)
cm25 <- table(bin_test$label, knn.25)
cm25
```

```
##    knn.25
##       0   1
##   0 146   7
##   1   2 145
```

```
mc_err25 <- mean(knn.25 != bin_test$label)
acc25 <- (1-mc_err25)
cat("Accuracy with k=25 is:",percent(acc25))
```

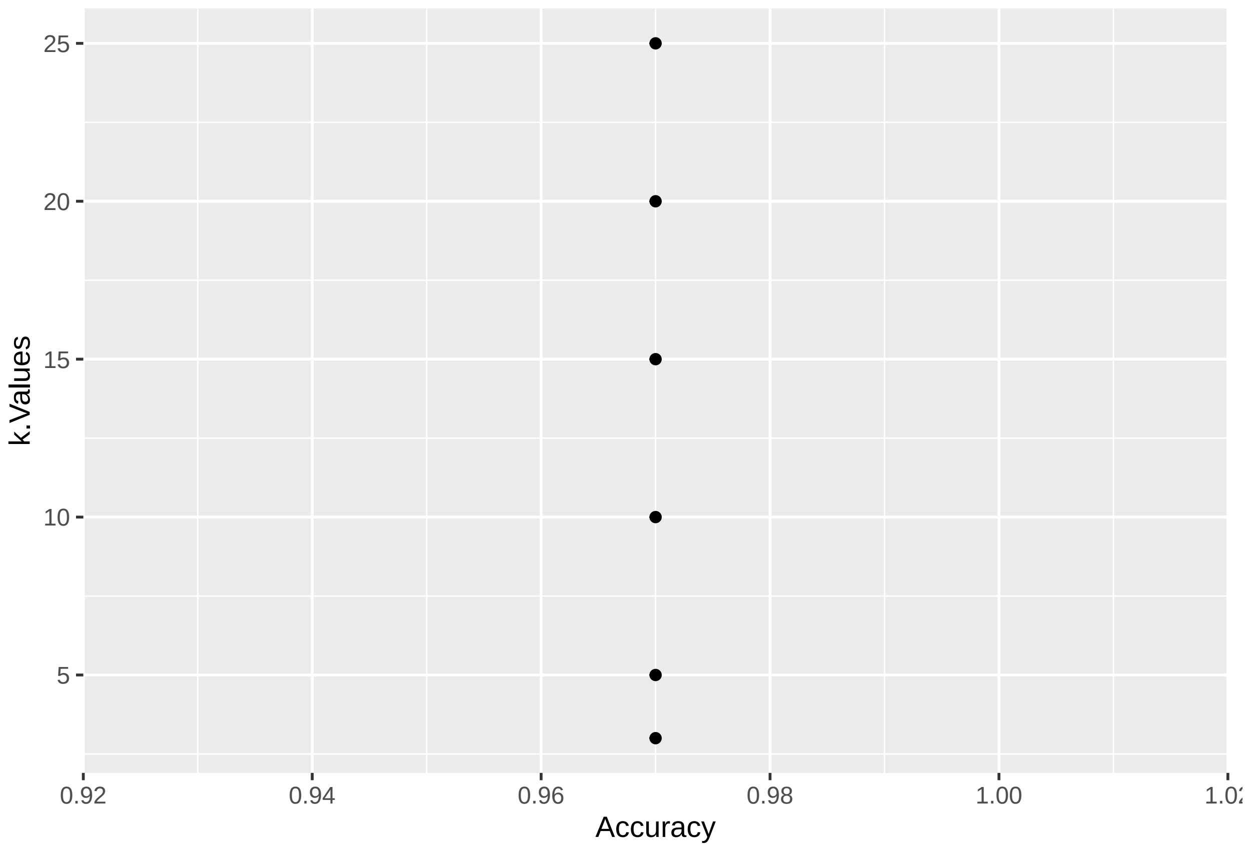
```
## Accuracy with k=25 is: 97%
```

Plot the accuracy and k values

```
k_vals <- c(3,5,10,15,20,25)
acc_list <- ls(pattern="acc\\d")
acc_vals <- sapply(acc_list, function(x) parse(text=x))
plot_vals <- as.data.frame(cbind(unlist(data.frame(as.list(acc_vals))), k_vals))
colnames(plot_vals) <- c("Accuracy", "k Values")
plot_vals <- transform(plot_vals, Accuracy = as.numeric(Accuracy))
plot_vals <- transform(plot_vals, Accuracy = round(Accuracy, digits=2))
plot_vals
```

```
##       Accuracy k.Values
## acc03     0.97        3
## acc05     0.97        5
## acc10     0.97       10
## acc15     0.97       15
## acc20     0.97       20
## acc25     0.97       25
```

```
acc_plt <- ggplot(plot_vals, aes(x=Accuracy, y=`k.Values`))
acc_plt + geom_point()
```

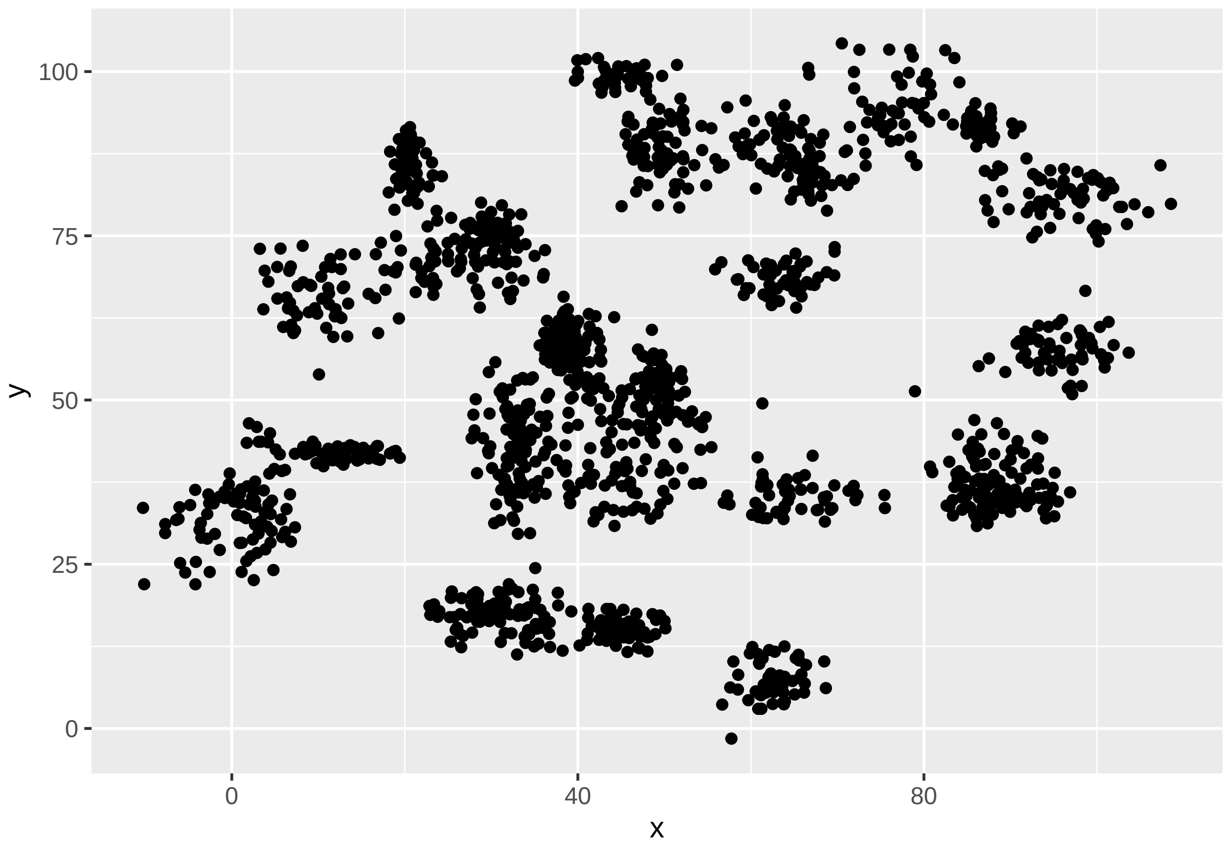


Trinary Data

```
trinary <- read.csv("/media/x/disk/School/DSC520/Datasets/trinary-classifier-data.csv")
head(trinary)
```

```
##   label        x        y
## 1     0 30.08387 39.63094
## 2     0 31.27613 51.77511
## 3     0 34.12138 49.27575
## 4     0 32.58222 41.23300
## 5     0 34.65069 45.47956
## 6     0 33.80513 44.24656
```

```
t_base <- ggplot(trinary, aes(x=x, y=y))
t_base + geom_point()
```



```
n_trin <- nrow(trinary)

# Set 80% of the rows for training sample
n_trin_train <- round(0.80 * n_trin)

# Create a vector of indices which is an 80% random sample
set.seed(1)
trin_train_indices <- sample(1:n_trin, n_trin_train)

# Subset the data frame to training indices only
trin_train <- trinary[trin_train_indices,]

#Exclude the training indices for test set
trin_test <- trinary[-trin_train_indices,]
```

```
knn.3 <- knn(train=trin_train, test=trin_test, cl=trin_train$label, k=3)
cm3 <- table(trin_test$label, knn.3)
cm3
```

```
##    knn.3
##       0   1   2
##   0  71   4   0
##   1   4 132   2
##   2   5   4  92
```

```
mc_err3 <- mean(knn.3 != trin_test$label)
acc3 <- percent(1-mc_err3)
```

```
cat("Accuracy with k=3 is:",acc3)
```

```
## Accuracy with k=3 is: 94%
```

```
knn.5 <- knn(train=trin_train, test=trin_test, cl=trin_train$label, k=5)
cm5 <- table(trin_test$label, knn.5)
cm5
```

```
##    knn.5
##       0   1   2
##   0  71   4   0
##   1   3 135   0
##   2   6   1  94
```

```
mc_err5 <- mean(knn.5 != trin_test$label)
acc5 <- percent(1-mc_err5)
cat("Accuracy with k=5 is:",acc5)
```

```
## Accuracy with k=5 is: 96%
```

```
knn.10 <- knn(train=trin_train, test=trin_test, cl=trin_train$label, k=10)
cm10 <- table(trin_test$label, knn.10)
cm10
```

```
##    knn.10
##       0   1   2
##   0  67   7   1
##   1   4 134   0
##   2   7   2  92
```

```
mc_err10 <- mean(knn.10 != trin_test$label)
acc10 <- percent(1-mc_err10)
cat("Accuracy with k=10 is:",acc10)
```

```
## Accuracy with k=10 is: 93%
```

```
knn.15 <- knn(train=trin_train, test=trin_test, cl=trin_train$label, k=15)
cm15 <- table(trin_test$label, knn.15)
cm15
```

```
##    knn.15
##       0   1   2
##   0  65   9   1
##   1   6 130   2
##   2   9   3  89
```

```
mc_err15 <- mean(knn.15 != trin_test$label)
acc15 <- percent(1-mc_err15)
cat("Accuracy with k=15 is:",acc15)
```

```
## Accuracy with k=15 is: 90%
```

```
knn.20 <- knn(train=trin_train, test=trin_test, cl=trin_train$label, k=20)
cm20 <- table(trin_test$label, knn.20)
cm20
```

```
##    knn.20
##       0   1   2
##   0  64  11   0
##   1   5 132   1
##   2   8   3  90
```

```
mc_err20 <- mean(knn.20 != trin_test$label)
acc20 <- percent(1-mc_err20)
cat("Accuracy with k=20 is:",acc20)
```

```
## Accuracy with k=20 is: 91%
```

```
knn.25 <- knn(train=trin_train, test=trin_test, cl=trin_train$label, k=25)
cm25 <- table(trin_test$label, knn.25)
cm25
```

```
##    knn.25
##       0   1   2
##   0  64  10   1
##   1   7 131   0
##   2   9   3  89
```

```
mc_err25 <- mean(knn.25 != trin_test$label)
acc25 <- percent(1-mc_err25)
cat("Accuracy with k=25 is:",acc25)
```

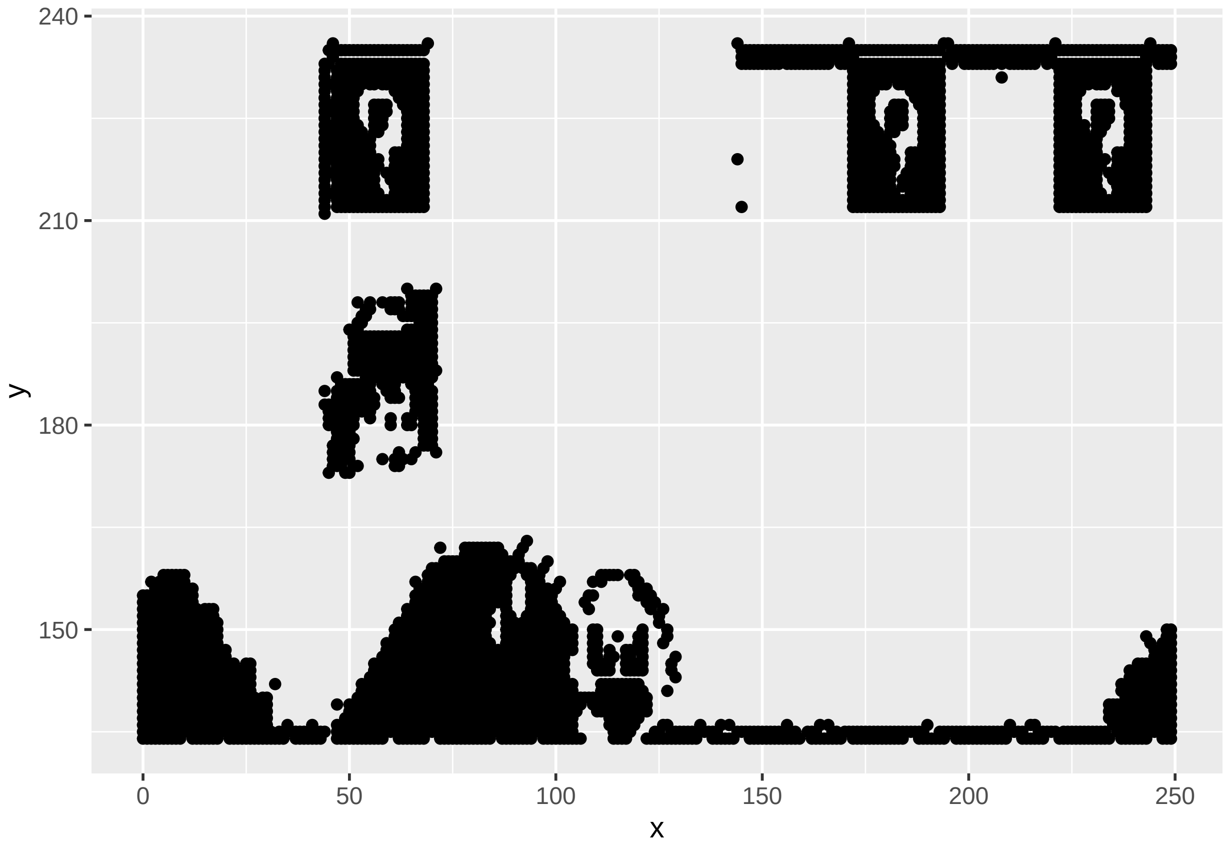
```
## Accuracy with k=25 is: 90%
```

Clustering

```
cluster <- read.csv("/media/x/disk/School/DSC520/Datasets/clustering-data.csv")
head(cluster)
```

```
##     x   y
## 1  46 236
## 2  69 236
## 3 144 236
## 4 171 236
## 5 194 236
## 6 195 236
```

```
cluster_plot <- ggplot(data=cluster, aes(x=x,y=y))

cluster_plot + geom_point()
```



```
set.seed(278613)
for(x in 2:12){
  print(paste0("Creating variable k",x))
  assign(paste0("k",x), kmeans(cluster, centers=x))
}
```

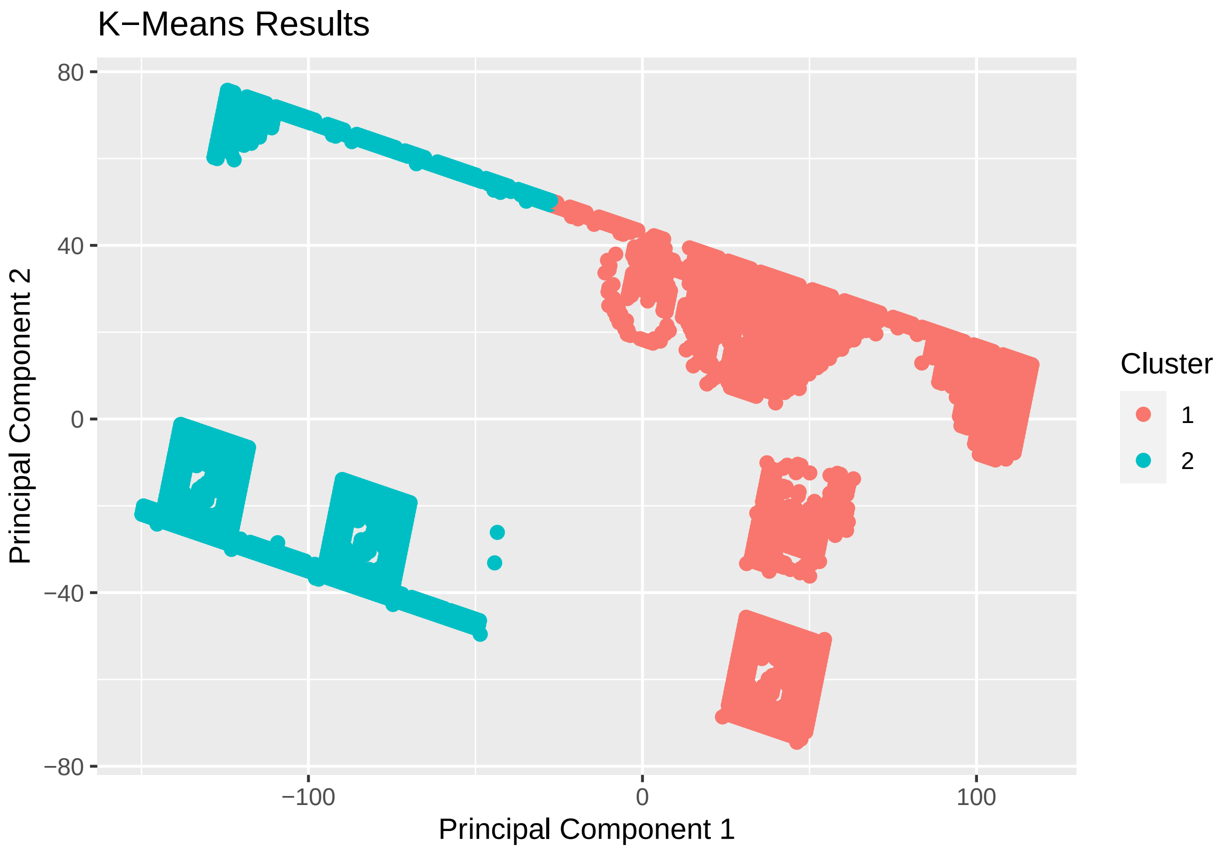
```
## [1] "Creating variable k2"
## [1] "Creating variable k3"
## [1] "Creating variable k4"
## [1] "Creating variable k5"
## [1] "Creating variable k6"
## [1] "Creating variable k7"
## [1] "Creating variable k8"
## [1] "Creating variable k9"
## [1] "Creating variable k10"
## [1] "Creating variable k11"
## [1] "Creating variable k12"
```

```
k2_cluster <- useful::plot.kmeans(k2, data=cluster)
k3_cluster <- useful::plot.kmeans(k3, data=cluster)
k4_cluster <- useful::plot.kmeans(k4, data=cluster)
k5_cluster <- useful::plot.kmeans(k5, data=cluster)
k6_cluster <- useful::plot.kmeans(k6, data=cluster)
k7_cluster <- useful::plot.kmeans(k7, data=cluster)
k8_cluster <- useful::plot.kmeans(k8, data=cluster)
k9_cluster <- useful::plot.kmeans(k9, data=cluster)
k10_cluster <- useful::plot.kmeans(k10, data=cluster)
```

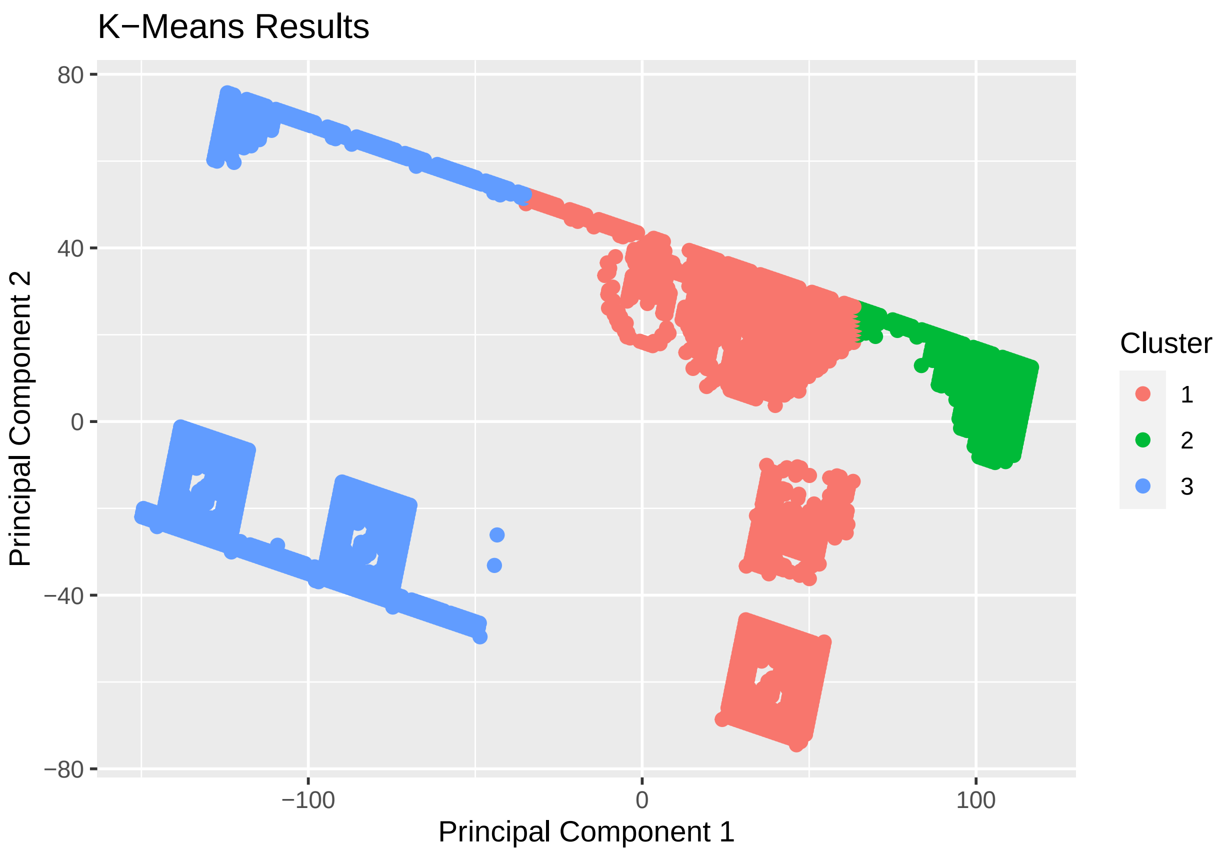
```
k11_cluster <- useful::plot.kmeans(k11, data=cluster)
k12_cluster <- useful::plot.kmeans(k12, data=cluster)
```

Display the plots

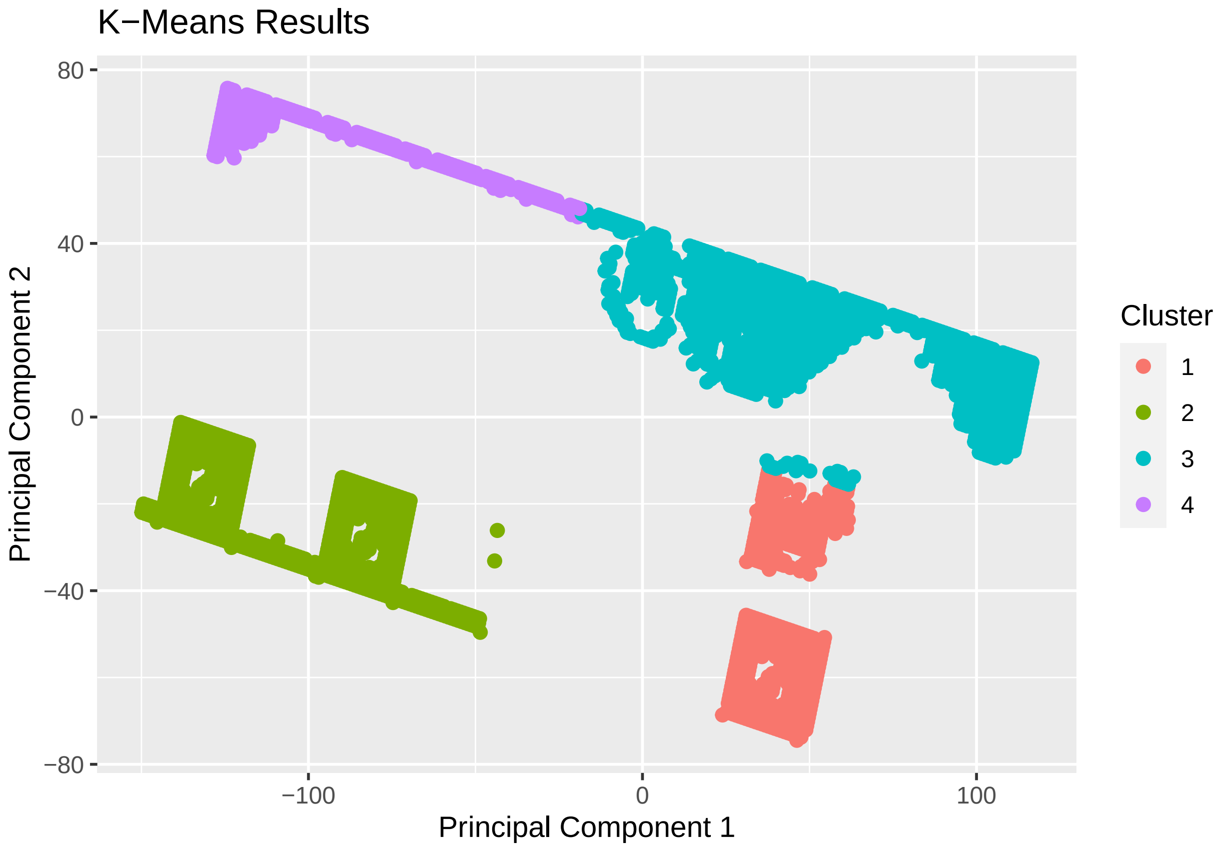
```
print(k2_cluster)
```



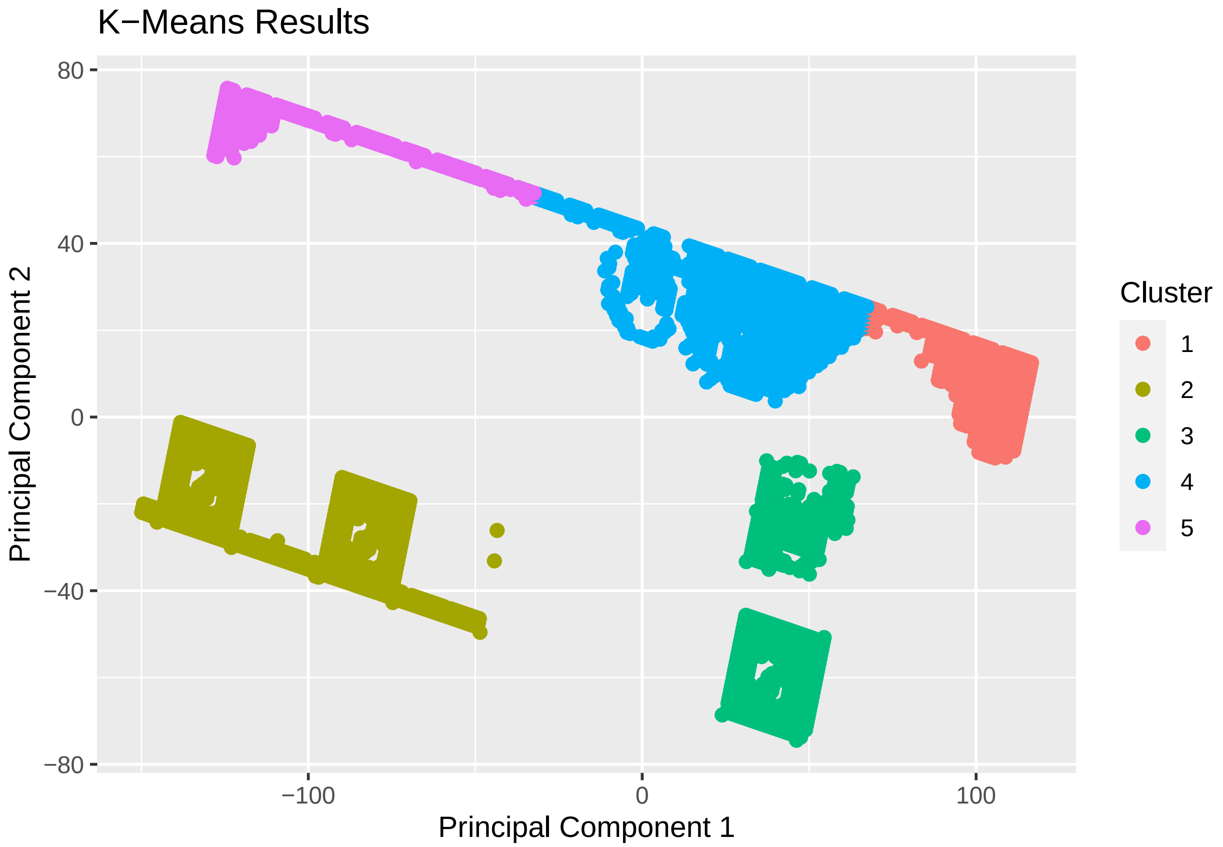
```
print(k3_cluster)
```



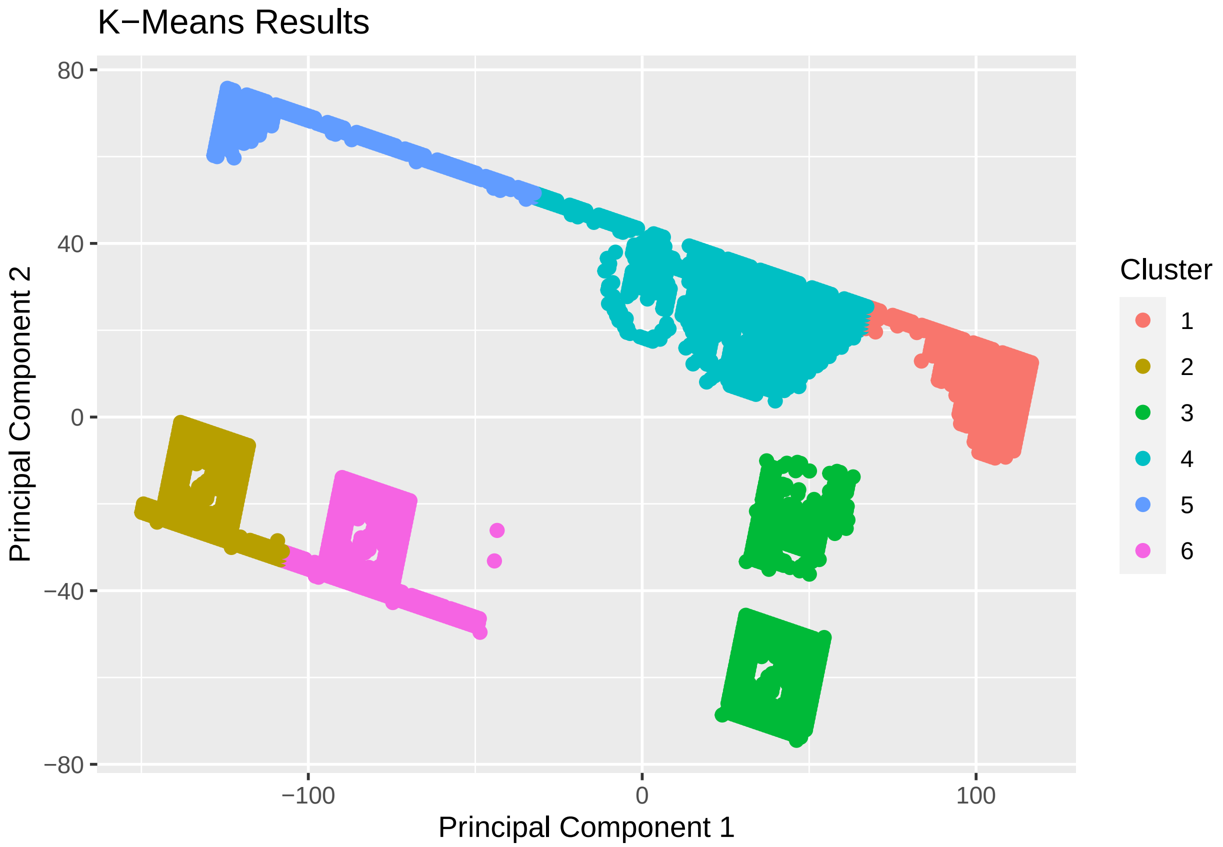
```
print(k4_cluster)
```



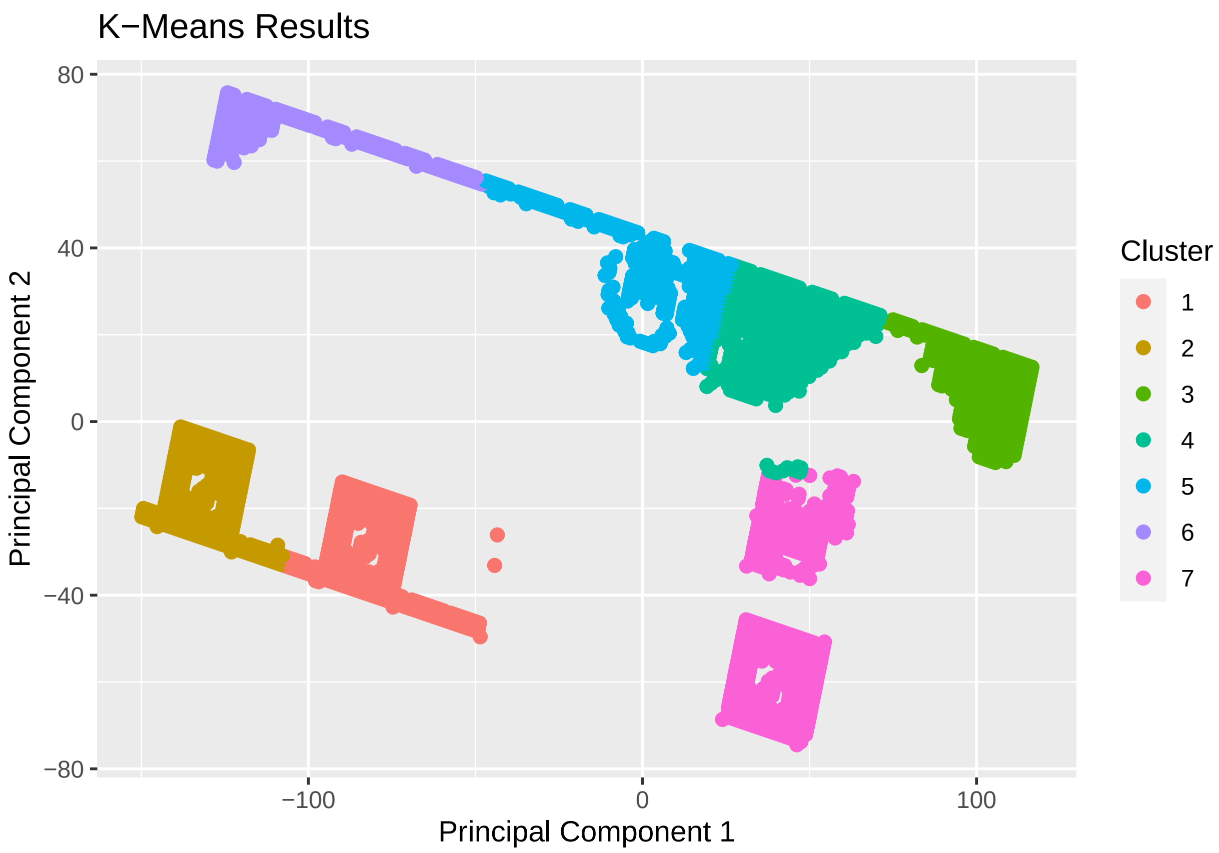
```
print(k5_cluster)
```



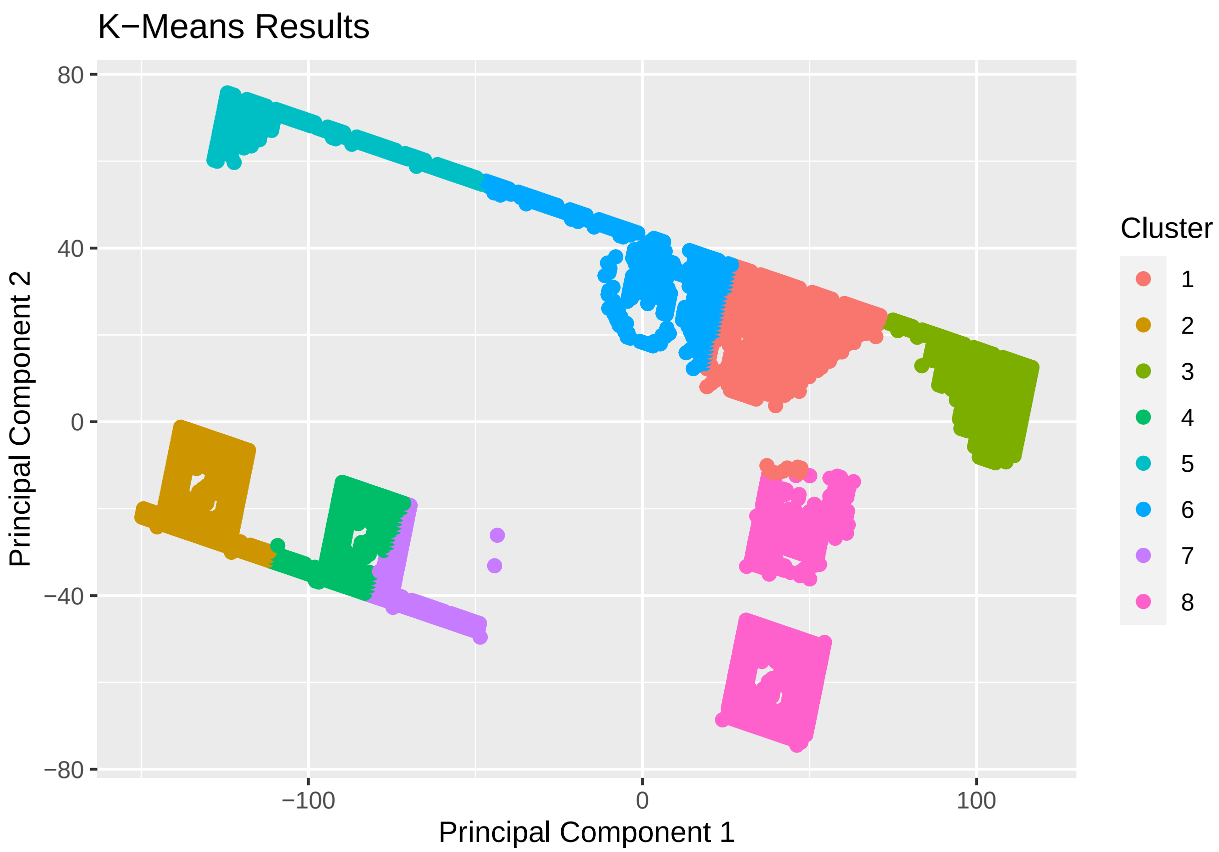
```
print(k6_cluster)
```



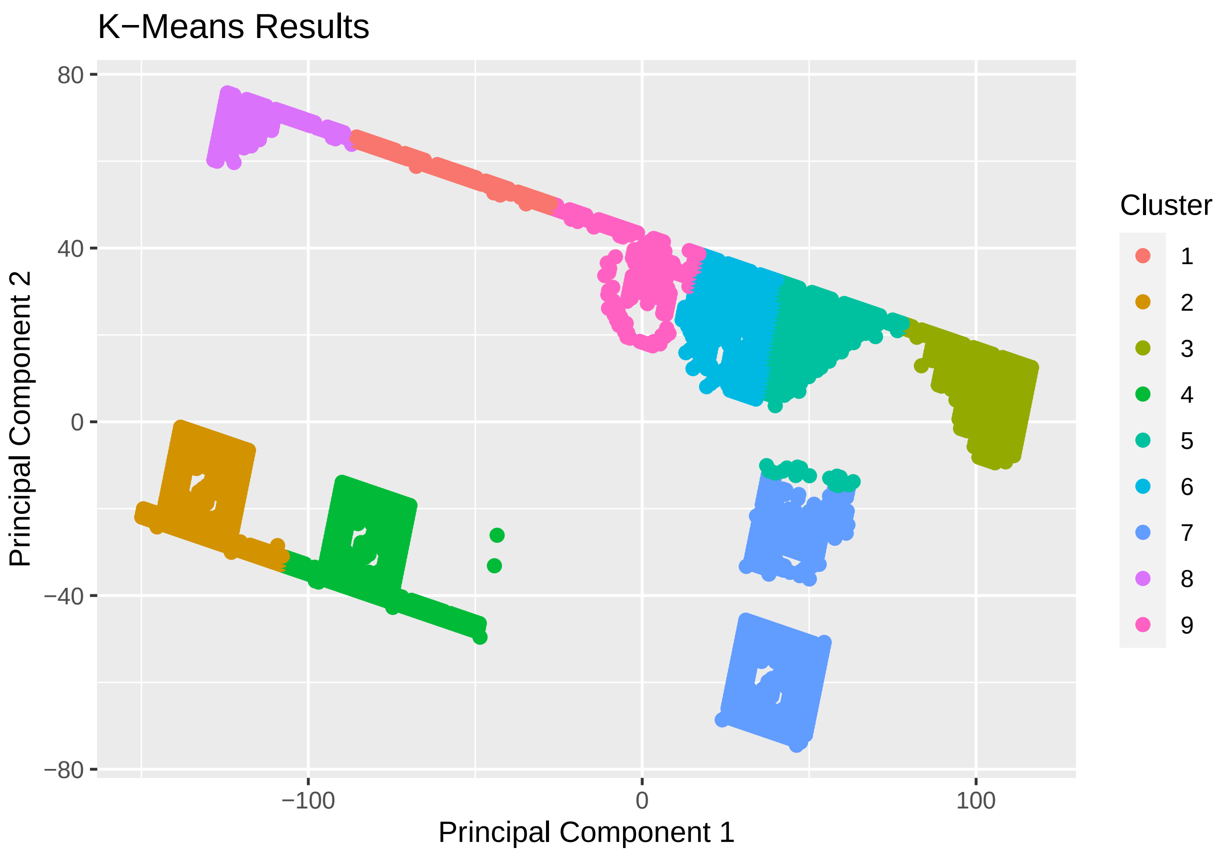
```
print(k7_cluster)
```



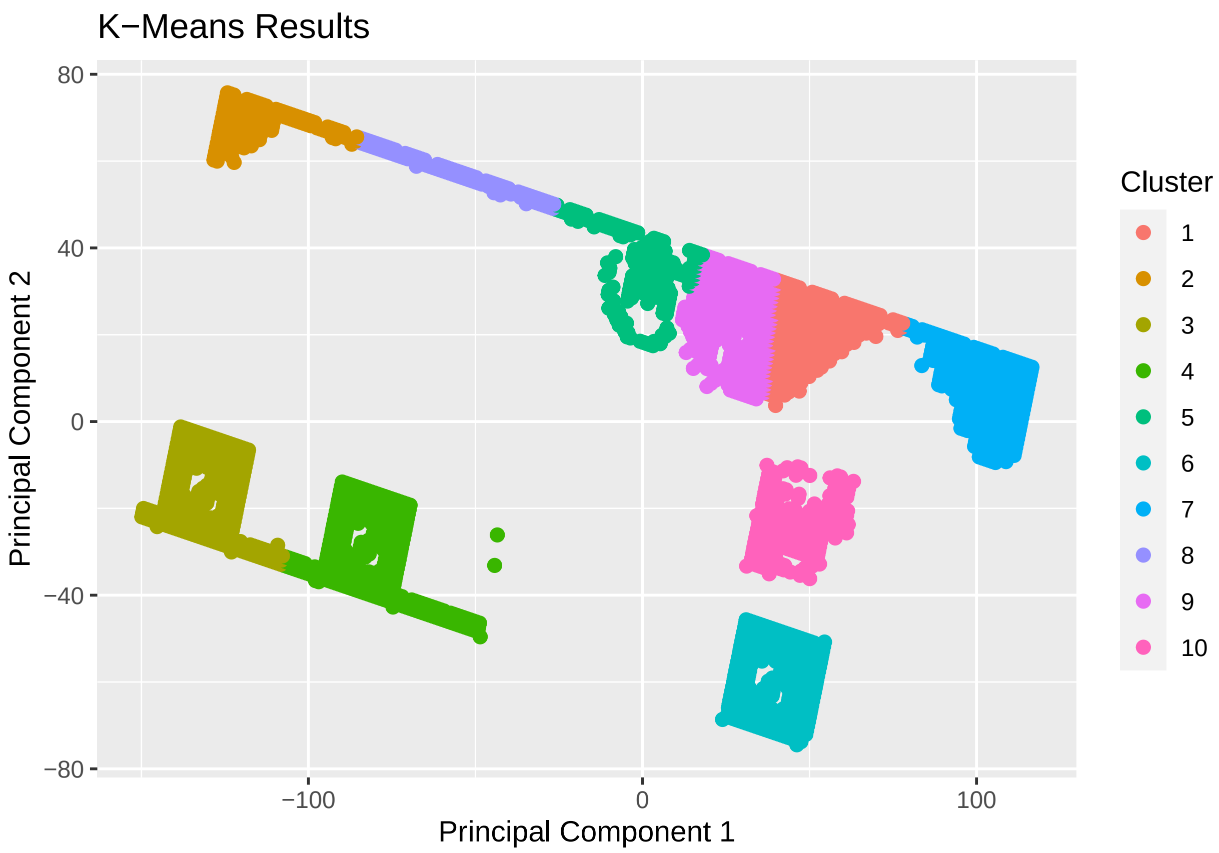
```
print(k8_cluster)
```



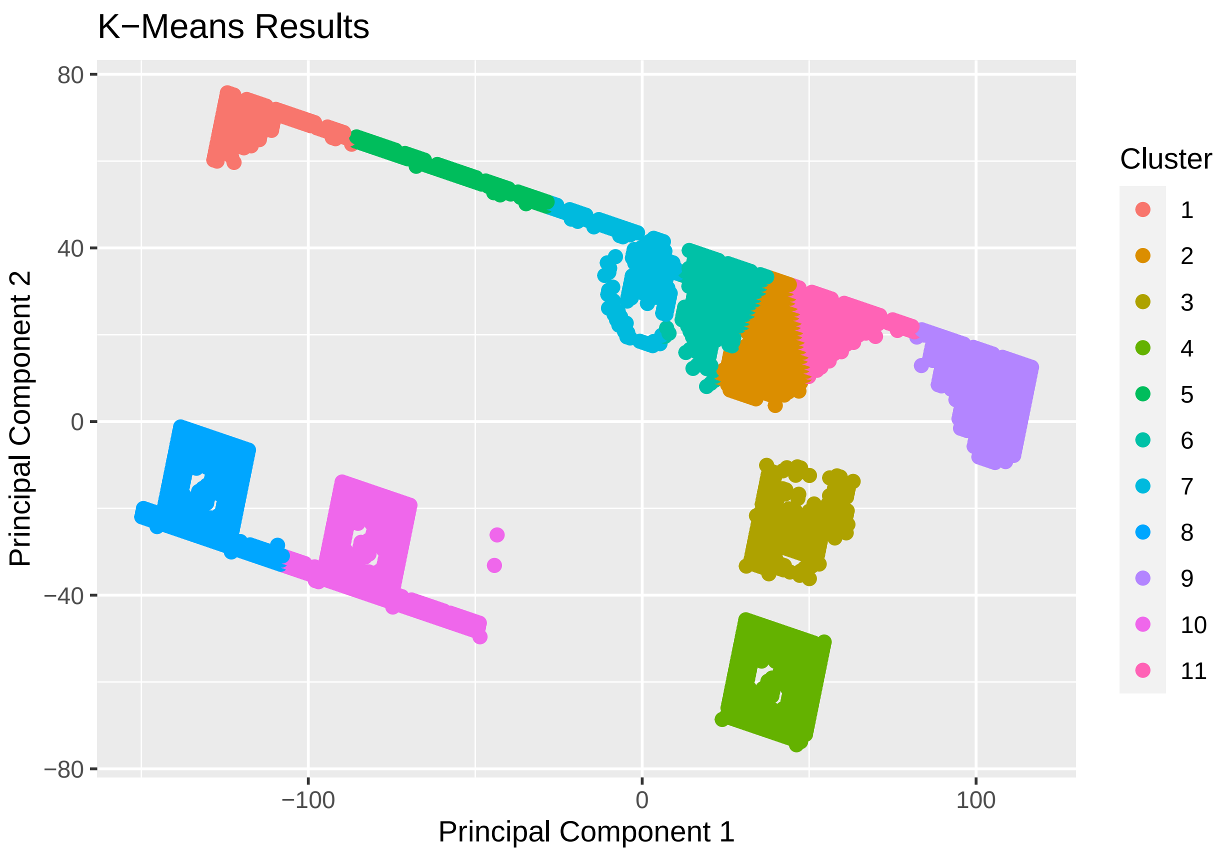
```
print(k9_cluster)
```



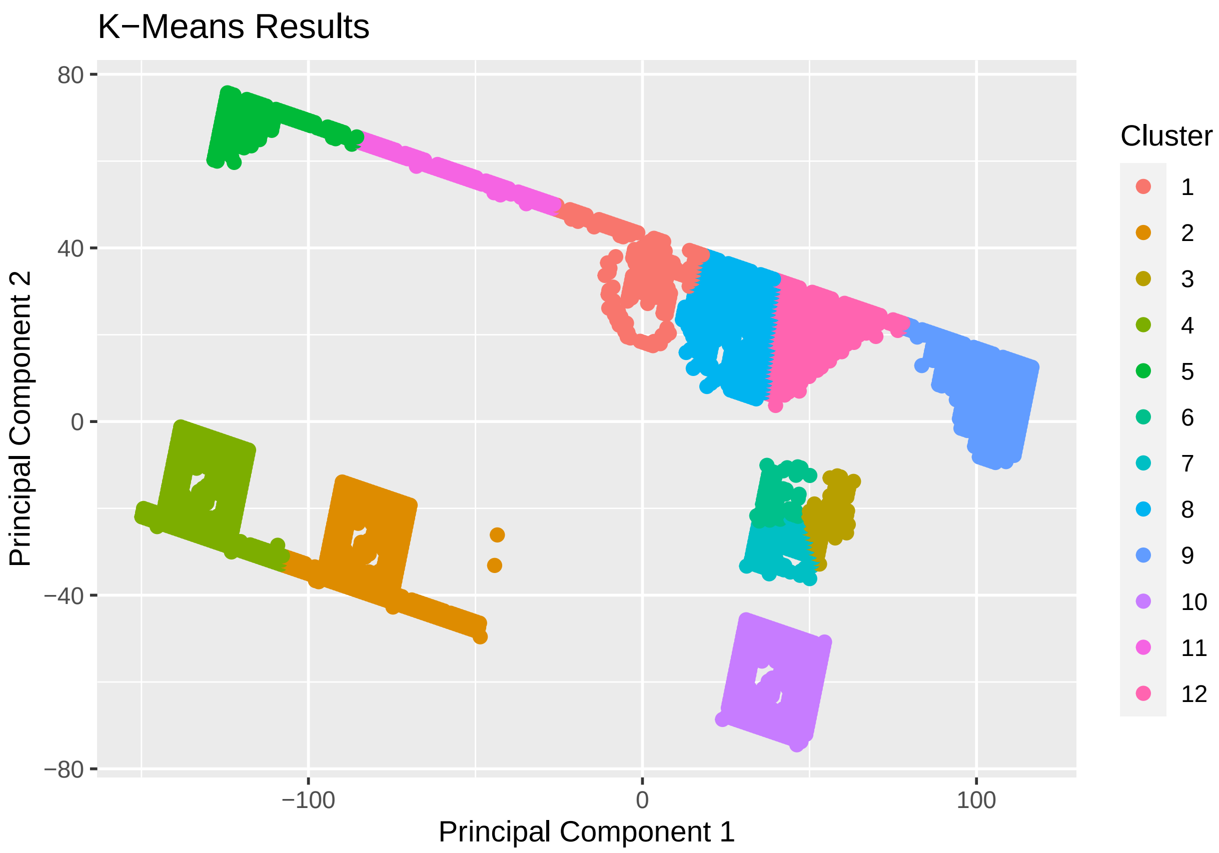
```
print(k10_cluster)
```



```
print(k11_cluster)
```



```
print(k12_cluster)
```



```
for(x in 2:12){
  temp_k <- eval(parse(text=paste0("k",x)), .GlobalEnv)
  print("Accuracy of")
  print(paste0("k",x))
  print(mean(temp_k$centers))
  cat("\n")
}
```

```
## [1] "Accuracy of"
## [1] "k2"
## [1] 158.9452
##
## [1] "Accuracy of"
## [1] "k3"
## [1] 135.8098
##
## [1] "Accuracy of"
## [1] "k4"
## [1] 157.1255
##
## [1] "Accuracy of"
## [1] "k5"
## [1] 144.1224
##
## [1] "Accuracy of"
## [1] "k6"
## [1] 156.056
##
## [1] "Accuracy of"
## [1] "k7"
## [1] 151.5478
##
## [1] "Accuracy of"
## [1] "k8"
## [1] 157.7385
##
## [1] "Accuracy of"
## [1] "k9"
## [1] 148.9338
##
## [1] "Accuracy of"
## [1] "k10"
## [1] 146.8534
##
## [1] "Accuracy of"
## [1] "k11"
## [1] 143.7182
##
## [1] "Accuracy of"
## [1] "k12"
## [1] 142.779
```

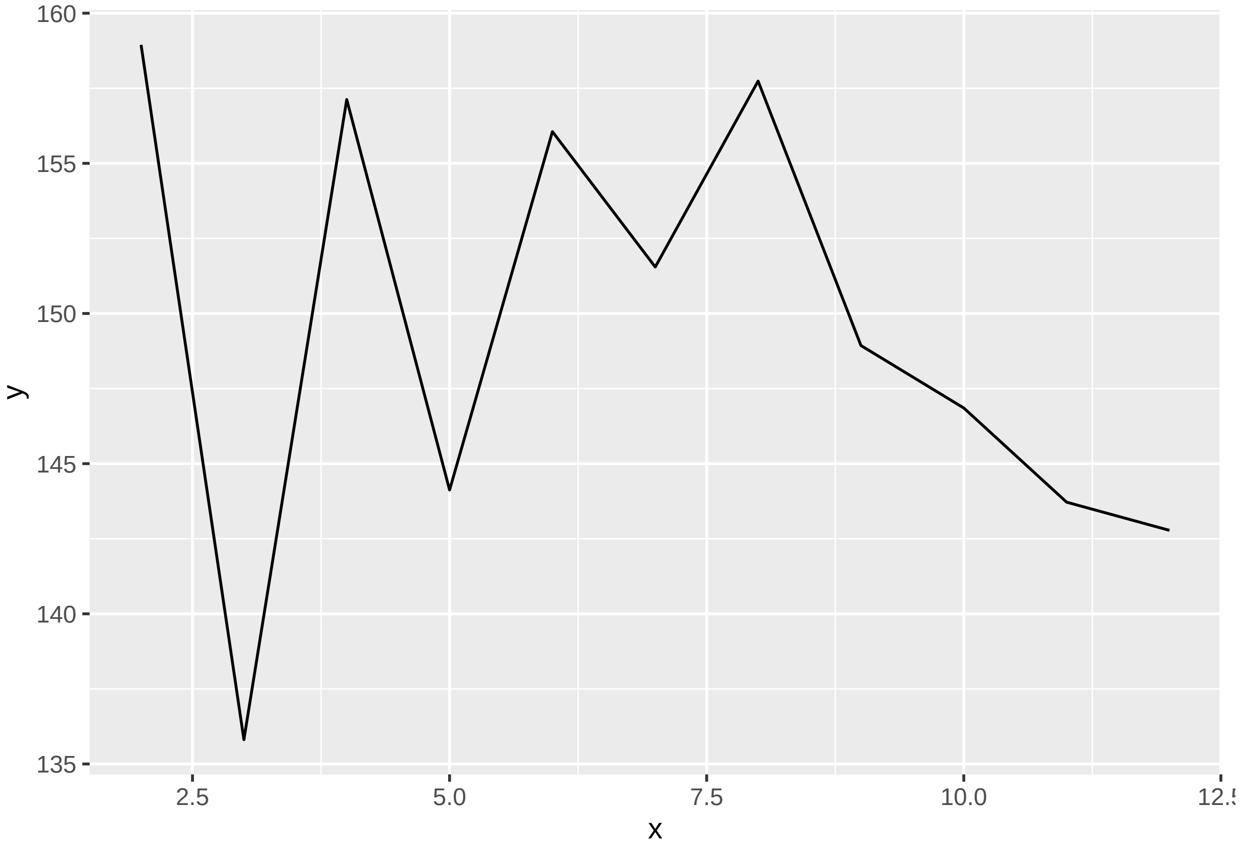
```
k_clusters <- list(k2,k3,k4,k5,k6,k7,k8,k9,k10,k11,k12)

k_dists <- sapply(k_clusters, function(x) mean(x$centers))
k_dists
```

```
##  [1] 158.9452 135.8098 157.1255 144.1224 156.0560 151.5478 157.7385 148.9338
##  [9] 146.8534 143.7182 142.7790
```

```
dist_data <- cbind(2:12, k_dists)
colnames(dist_data) <- c("x", "y")
dist_data <- data.frame(dist_data)
dist_data
```

```
##     x        y
## 1   2 158.9452
## 2   3 135.8098
## 3   4 157.1255
## 4   5 144.1224
## 5   6 156.0560
## 6   7 151.5478
## 7   8 157.7385
## 8   9 148.9338
## 9  10 146.8534
## 10 11 143.7182
## 11 12 142.7790
```

```
ggplot(dist_data, aes(x=x,y=y))+ geom_line()
```



I would say that the “elbow” of this plot is around 8, or just after 7.5.