

# DSC520 Week10 Exercise 11.2.1

Venkat Jagadeesh Jampani

March 5th 2022

```
# Load the packages
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.1 --

## v ggplot2 3.3.5      v purrr  0.3.4
## v tibble  3.1.6      v dplyr  1.0.7
## v tidyr   1.1.4      v stringr 1.4.0
## v readr   2.1.1      v forcats 0.5.1

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()

library(cluster)
library(factoextra)

## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa

library(caTools)

setwd("/Users/Jagadeesh/Documents/GitHub/dsc520")

# Load Clustering Data Dataset
cluster_df <- read.csv("data/clustering-data.csv")

# Check structure of clustering_data_df
str(cluster_df)

## 'data.frame':    4022 obs. of  2 variables:
## $ x: int  46 69 144 171 194 195 221 244 45 47 ...
## $ y: int  236 236 236 236 236 236 236 236 235 235 ...

# Check sample rows of clustering_data_df
head(cluster_df)

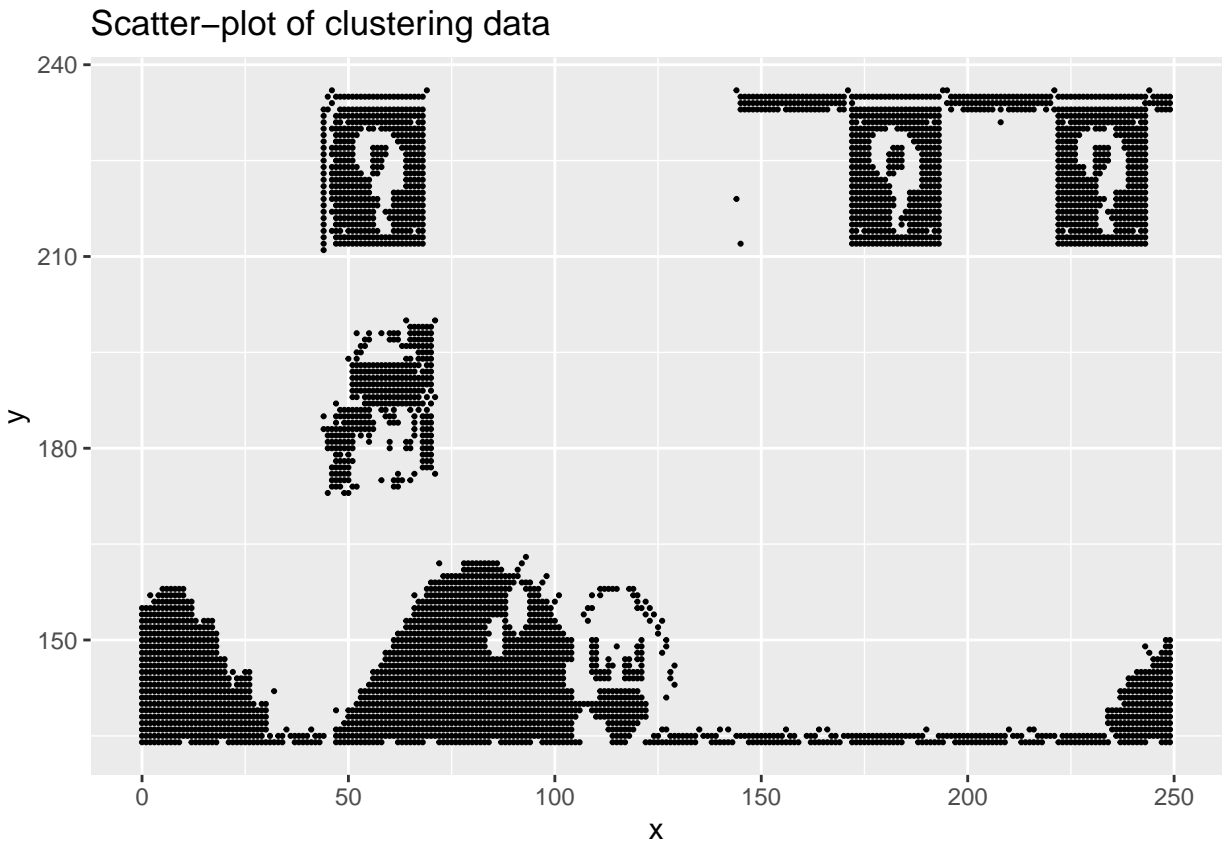
##      x      y
## 1  46 236
```

```
## 2 69 236
## 3 144 236
## 4 171 236
## 5 194 236
## 6 195 236
```

```
# i. Plot the data set using a scatter plot.
```

```
library(ggplot2)
```

```
ggplot(data=cluster_df,aes(x=x,y=y)) + geom_point(size=0.4) + ggtitle("Scatter-plot of clustering data")
```



```
# ii. Fit the dataset using the k-means algorithm from k=2 to k=12. Create a scatter plot of the results
```

```
set.seed(123)
```

```
kmeans_2 <- kmeans(cluster_df, 2, iter.max = 300, nstart = 10)
```

```
set.seed(123)
```

```
kmeans_3 <- kmeans(cluster_df, 3, iter.max = 300, nstart = 10)
```

```
set.seed(123)
```

```
kmeans_4 <- kmeans(cluster_df, 4, iter.max = 300, nstart = 10)
```

```
set.seed(123)
```

```
kmeans_5 <- kmeans(cluster_df, 5, iter.max = 300, nstart = 10)
```

```
set.seed(123)
```

```
kmeans_6 <- kmeans(cluster_df, 6, iter.max = 300, nstart = 10)
```

```
set.seed(123)
```

```
kmeans_7 <- kmeans(cluster_df, 7, iter.max = 300, nstart = 10)
```

```
set.seed(123)
```

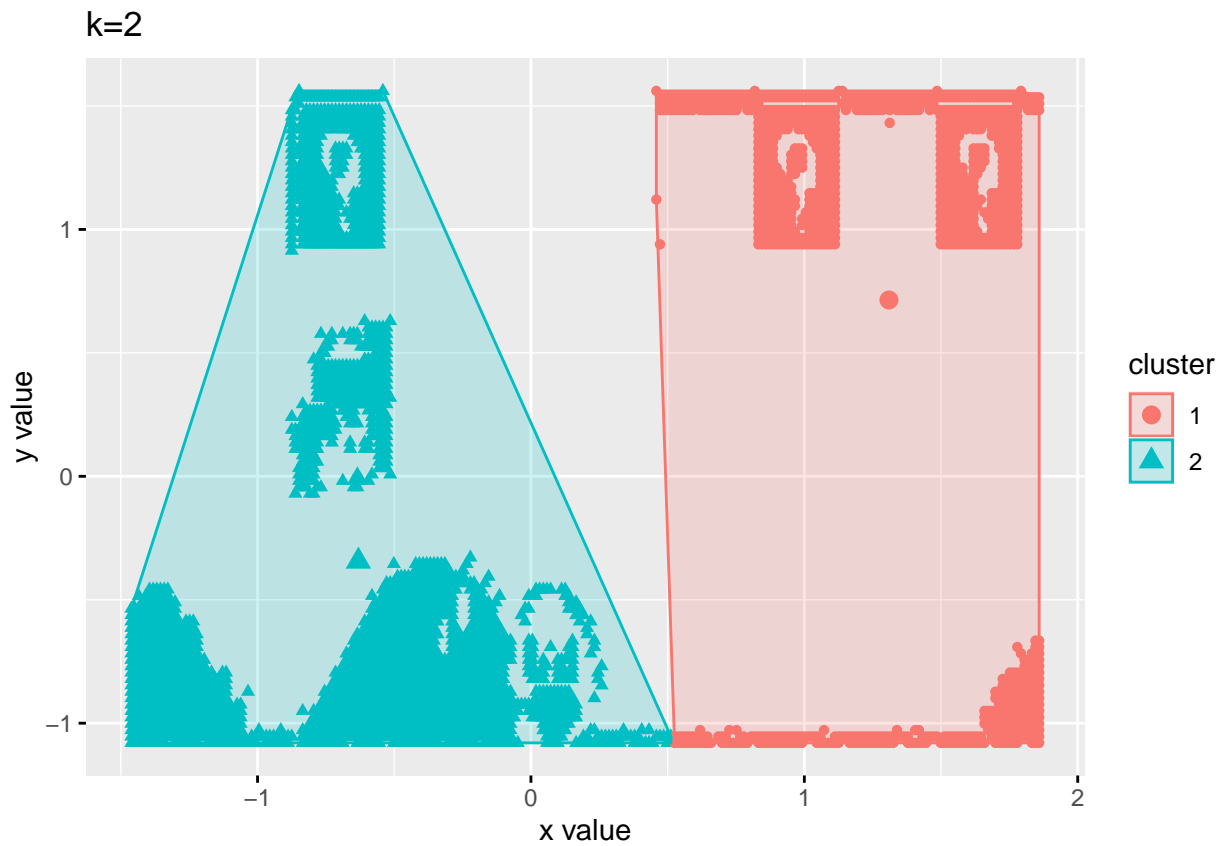
```
kmeans_8 <- kmeans(cluster_df, 8, iter.max = 300, nstart = 10)
```

```

set.seed(123)
kmeans_9 <- kmeans(cluster_df, 9, iter.max = 300, nstart = 10)
set.seed(123)
kmeans_10 <- kmeans(cluster_df, 10, iter.max = 300, nstart = 10)
set.seed(123)
kmeans_11 <- kmeans(cluster_df, 11, iter.max = 300, nstart = 10)
set.seed(123)
kmeans_12 <- kmeans(cluster_df, 12, iter.max = 300, nstart = 10)

# Plots to compare
fviz_cluster(kmeans_2, geom="point", data=cluster_df) + ggtitle("k=2")

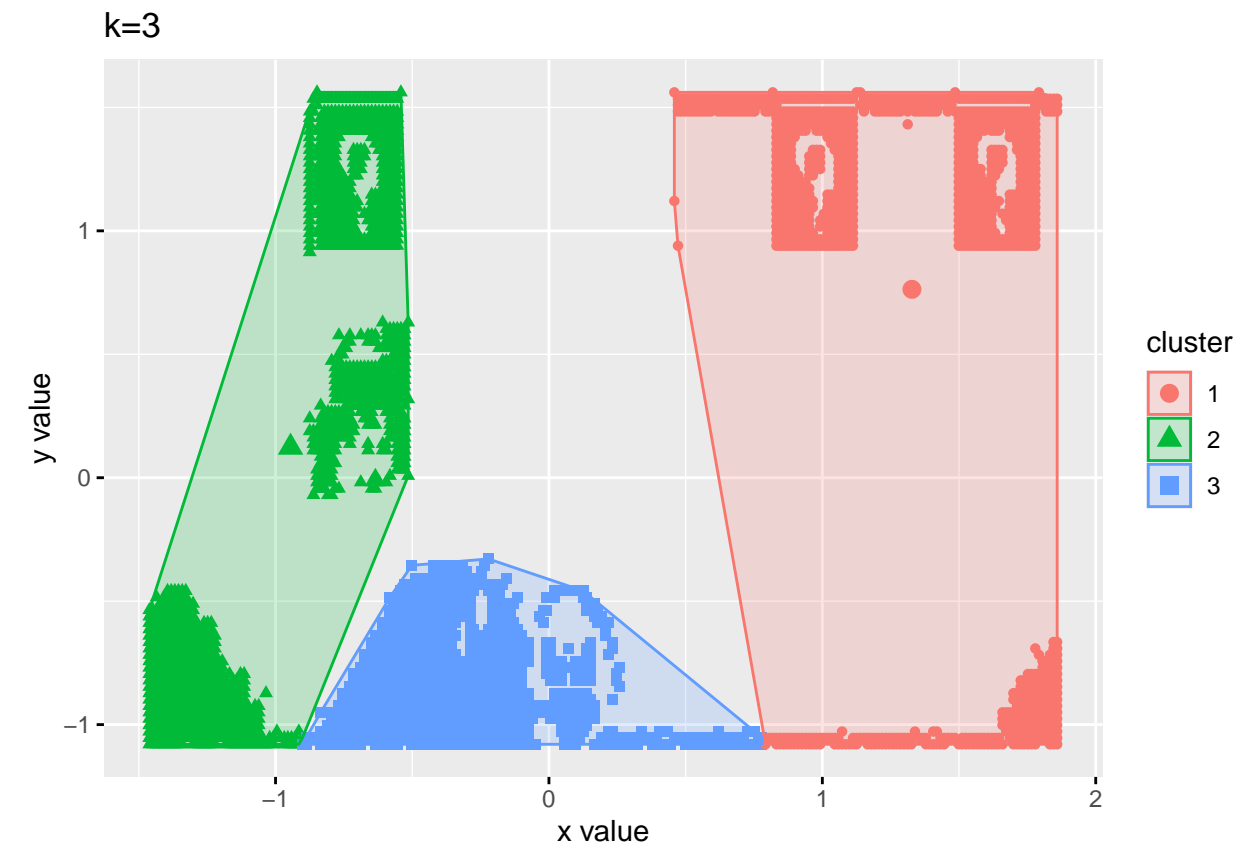
```



```

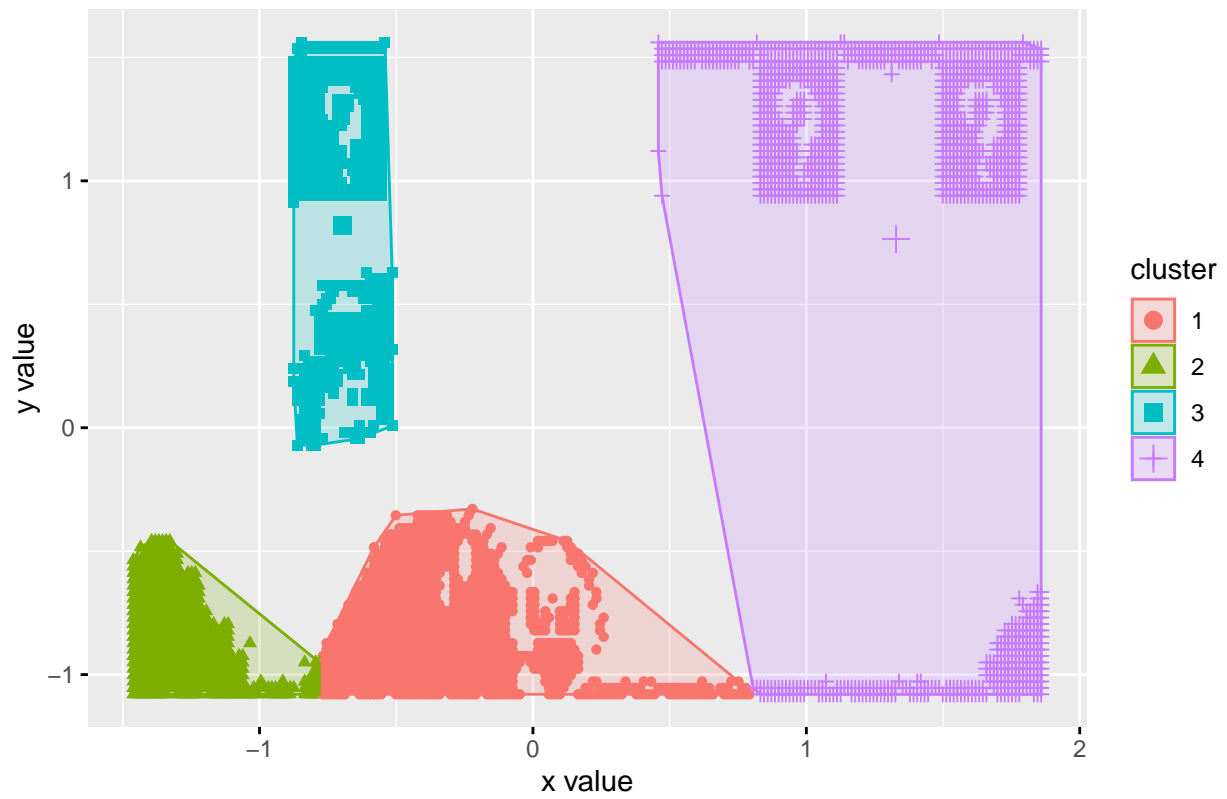
fviz_cluster(kmeans_3, geom="point", data=cluster_df) + ggtitle("k=3")

```



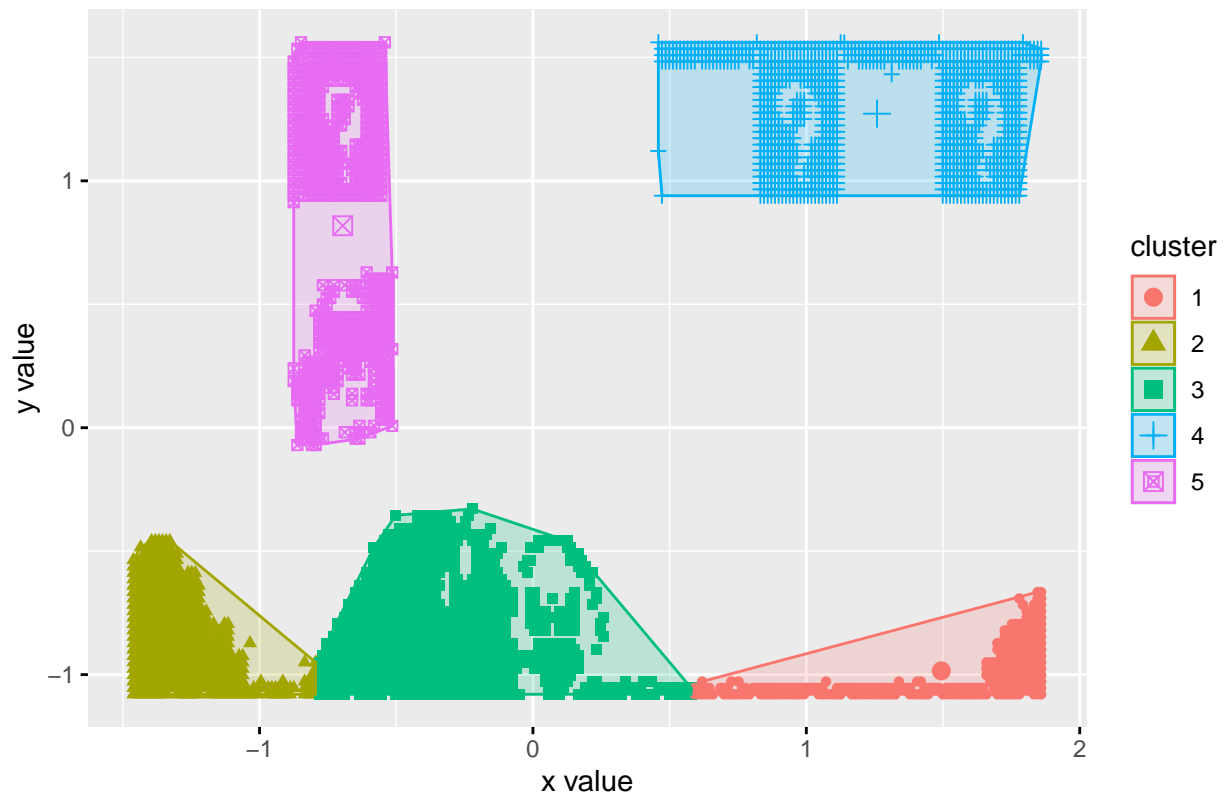
```
fviz_cluster(kmeans_4, geom="point", data=cluster_df) + ggtitle("k=4")
```

k=4

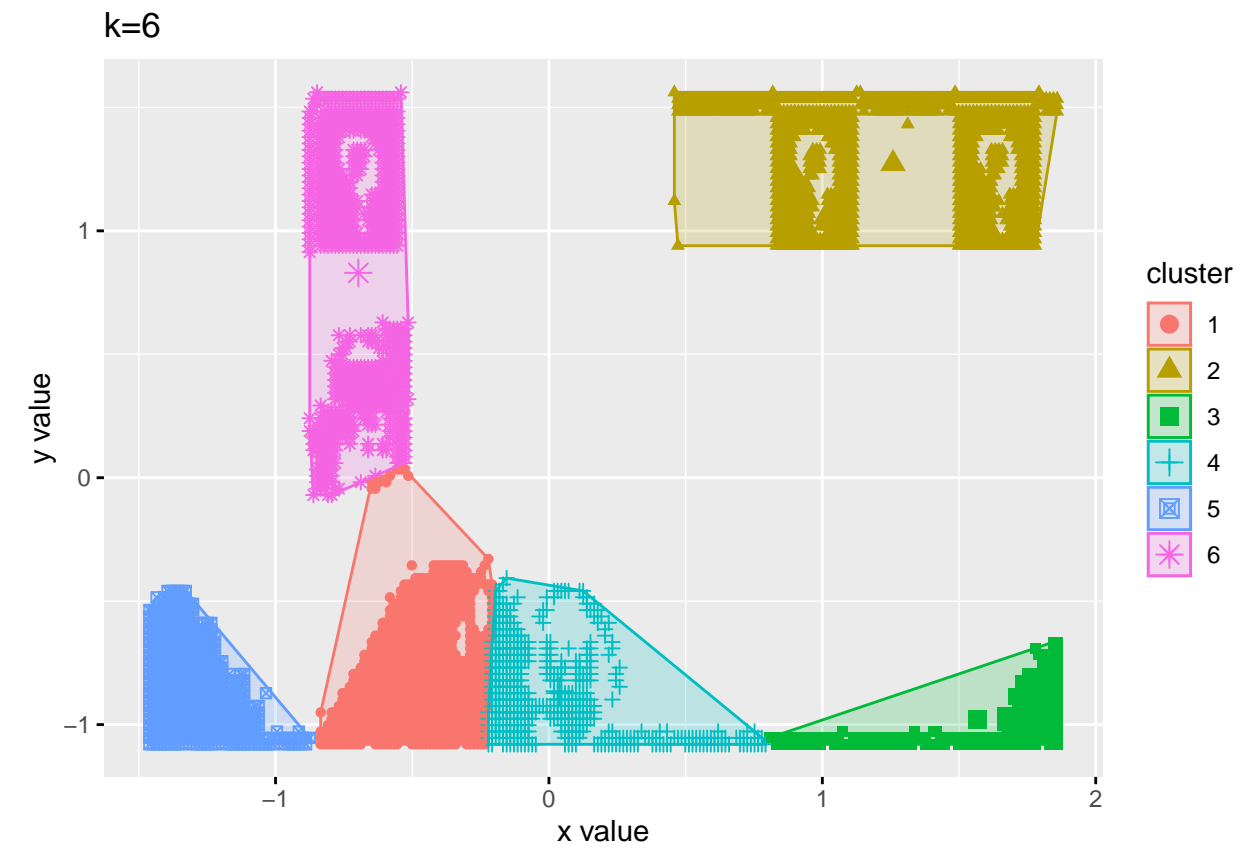


```
fviz_cluster(kmeans_5, geom="point", data=cluster_df) + ggtitle("k=5")
```

k=5

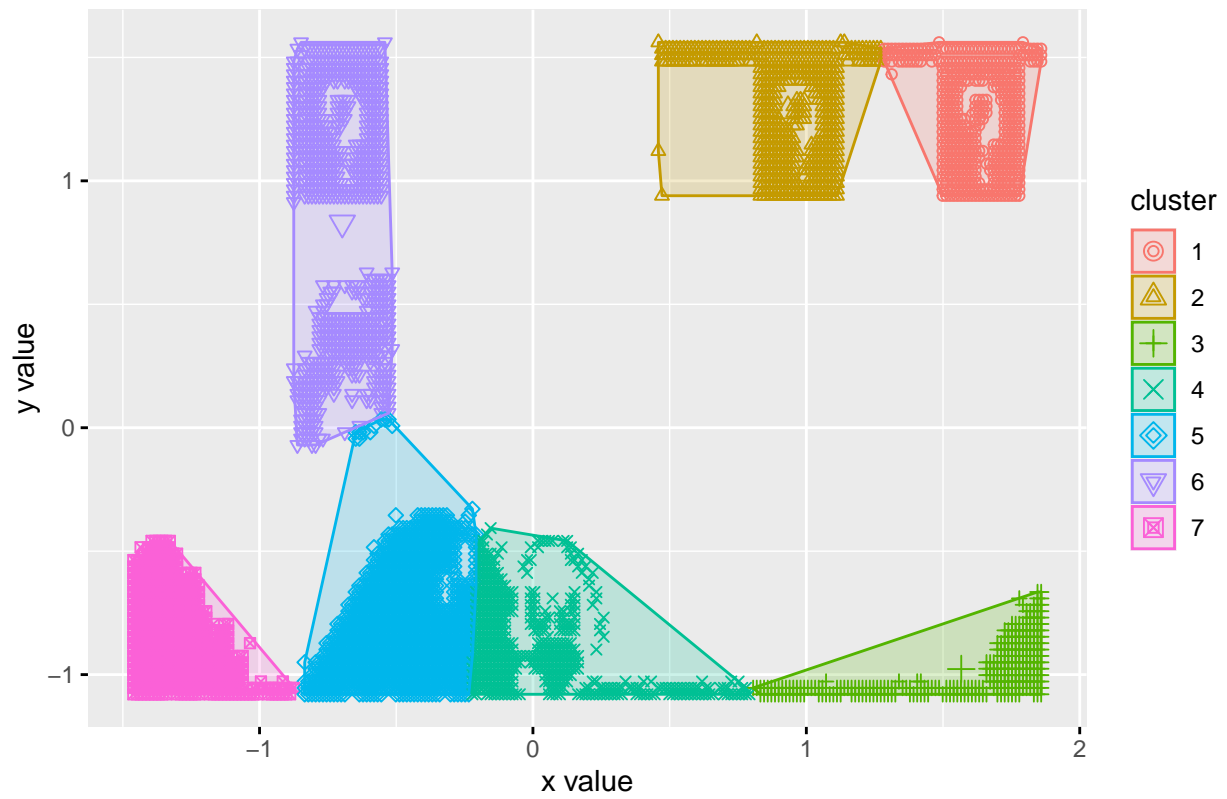


```
fviz_cluster(kmeans_6, geom="point", data=cluster_df) + ggtitle("k=6")
```



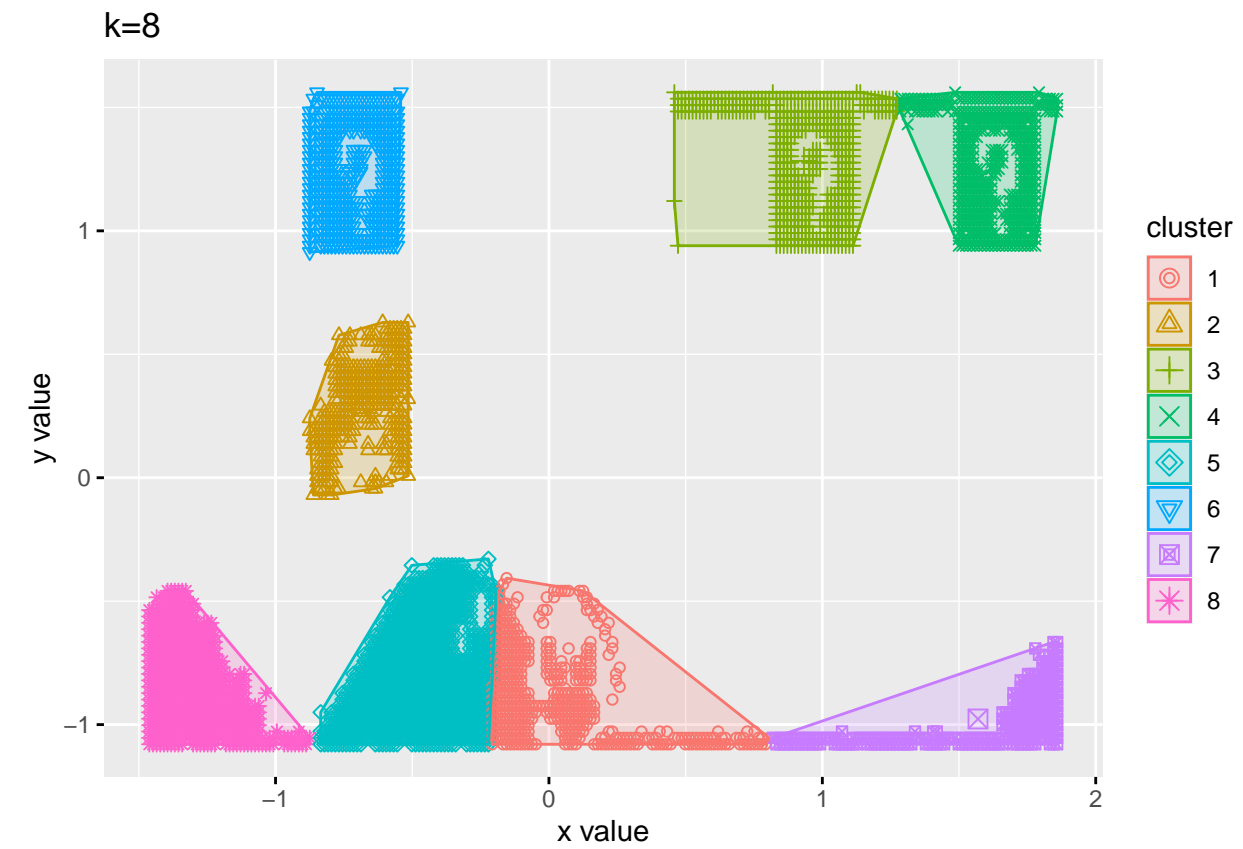
```
fviz_cluster(kmeans_7, geom="point", data=cluster_df) + ggtitle("k=7")
```

k=7

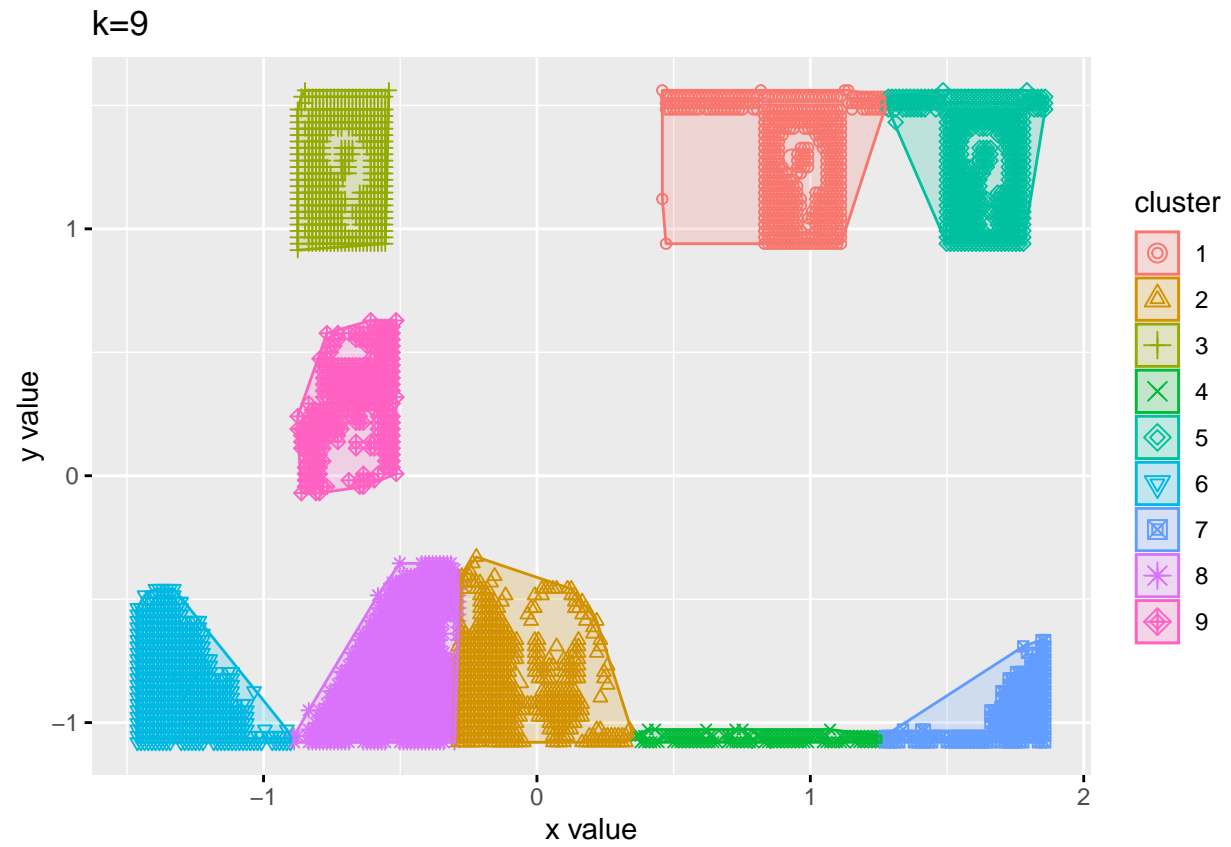


```
fviz_cluster(kmeans_8, geom="point", data=cluster_df) + ggtitle("k=8")
```



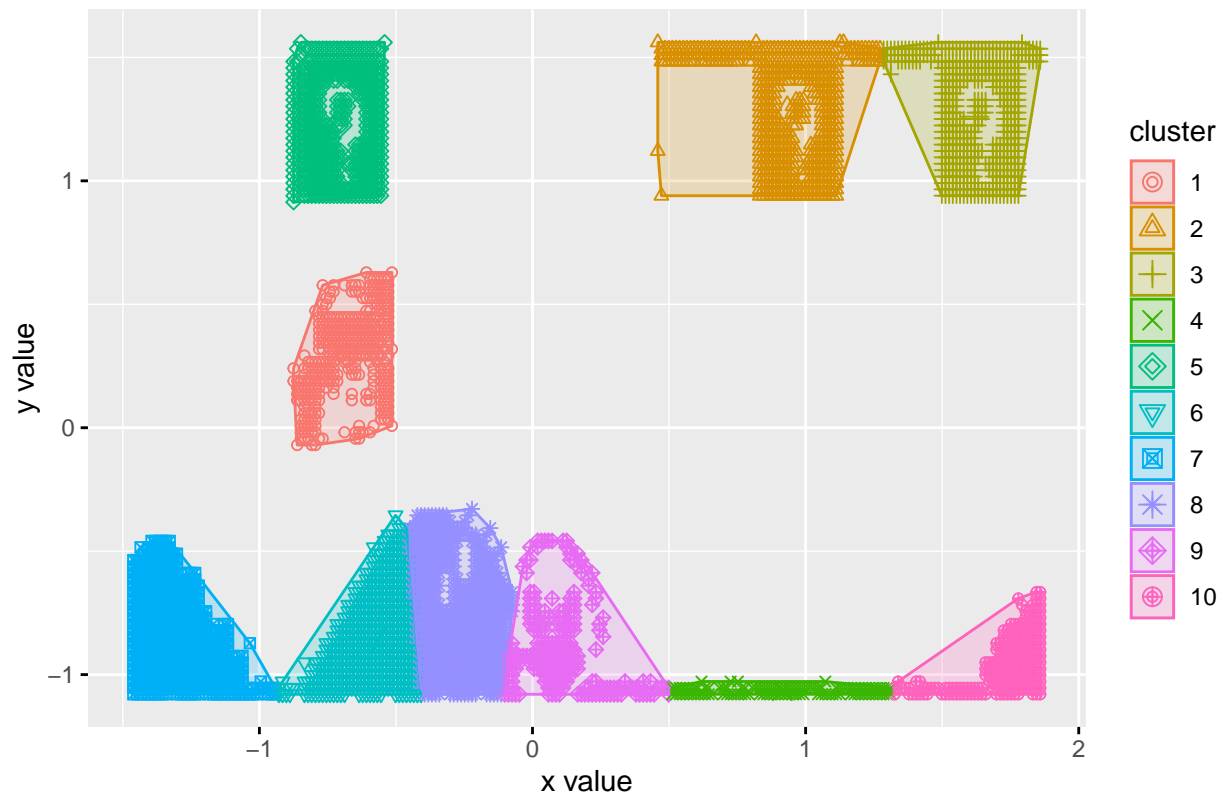


```
fviz_cluster(kmeans_9, geom="point", data=cluster_df) + ggtitle("k=9")
```



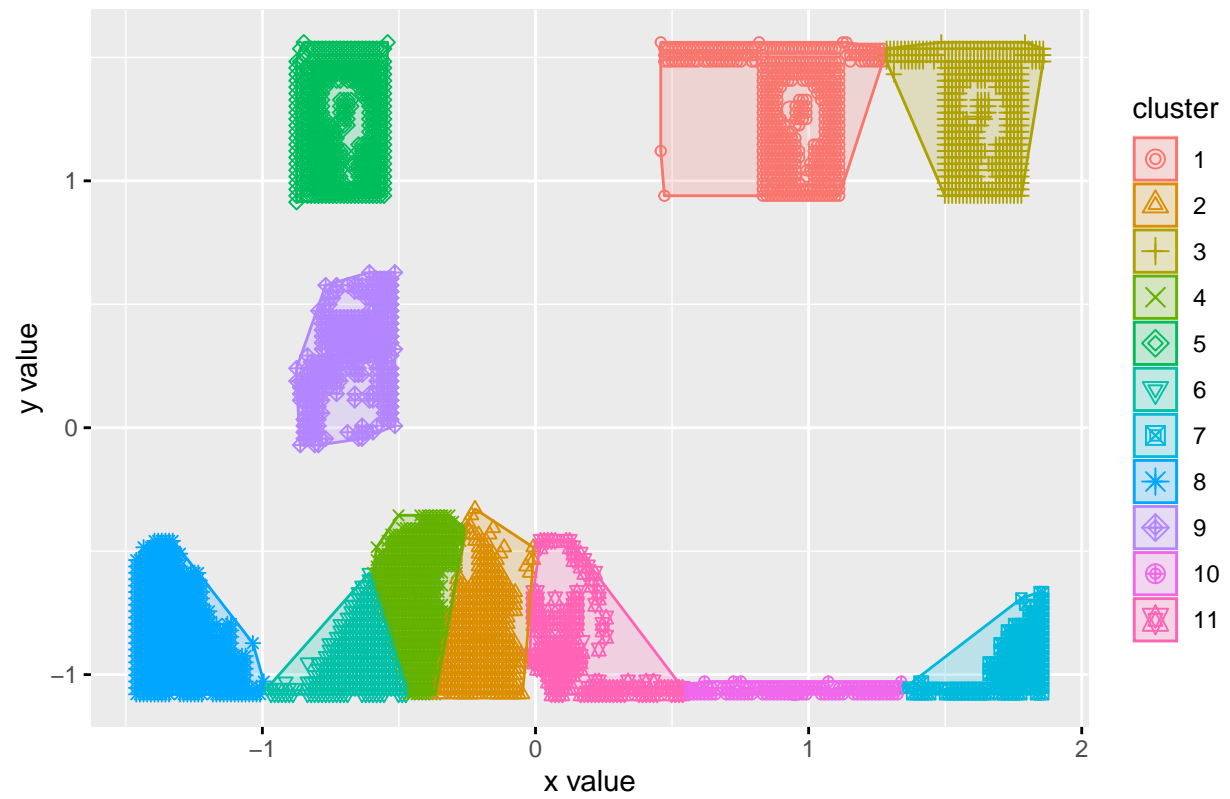
```
fviz_cluster(kmeans_10, geom="point", data=cluster_df) + ggtitle("k=10")
```

k=10



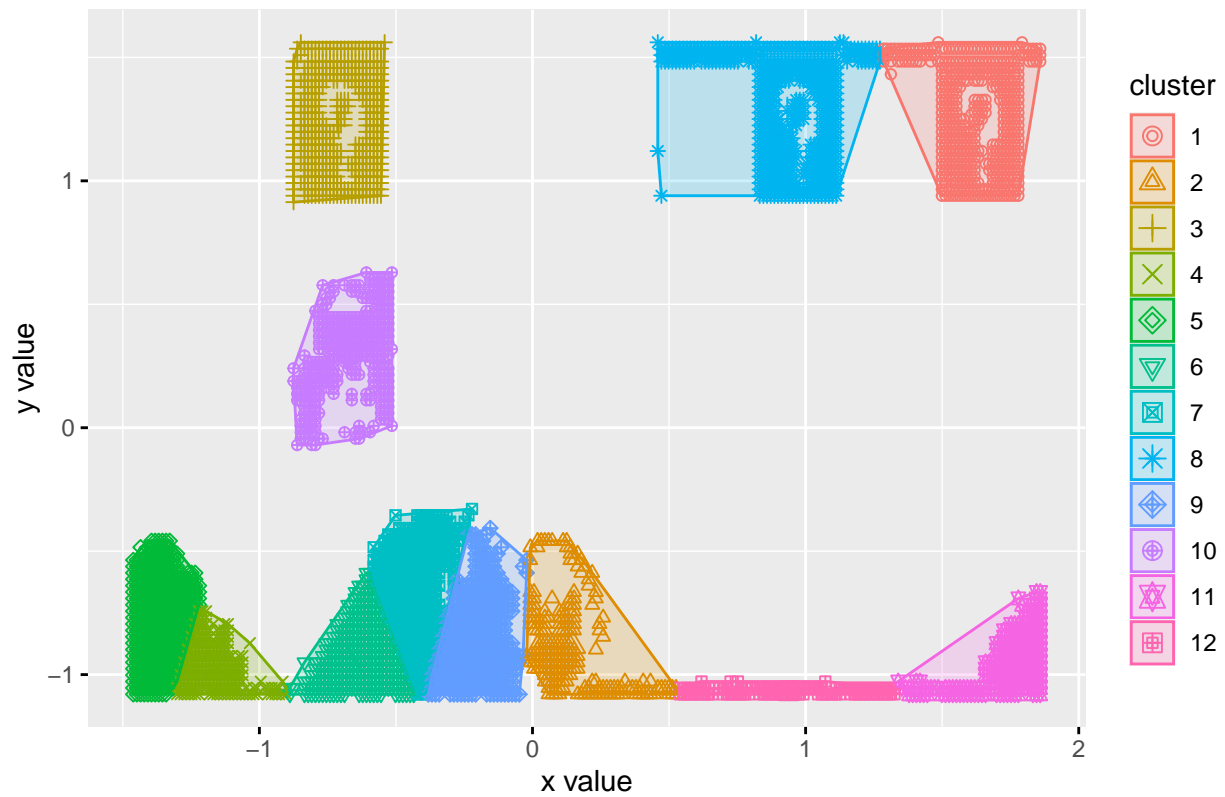
```
fviz_cluster(kmeans_11, geom="point", data=cluster_df) + ggtitle("k=11")
```

k=11



```
fviz_cluster(kmeans_12, geom="point", data=cluster_df) + ggtitle("k=12")
```

k=12

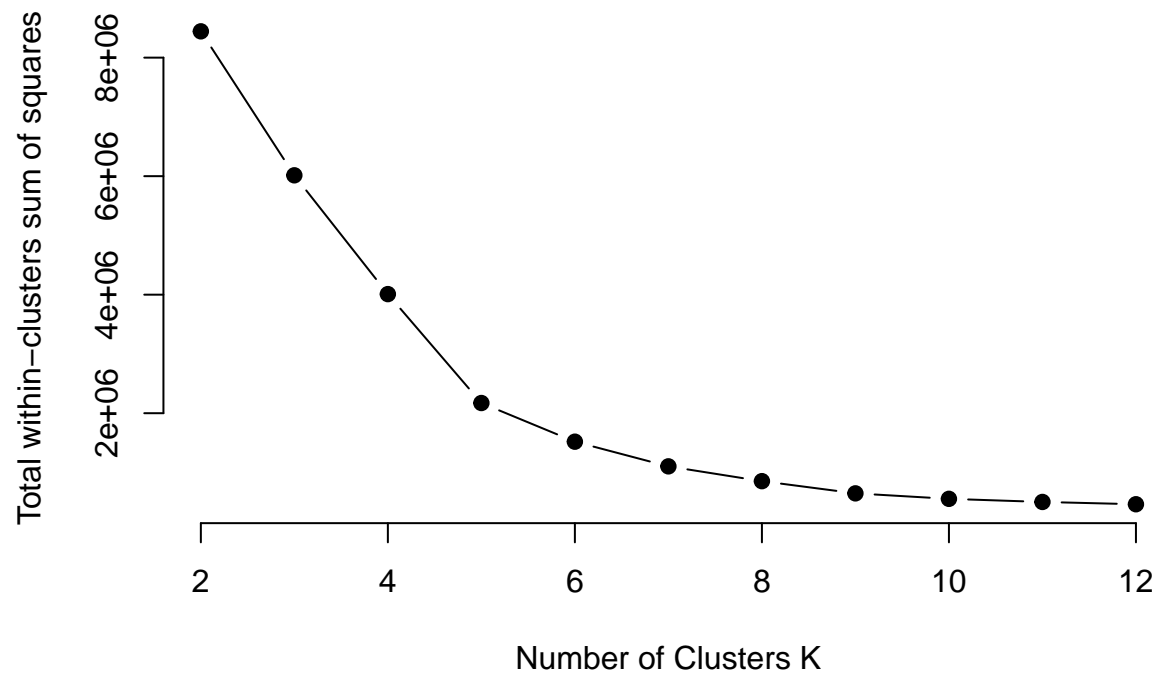


```
set.seed(123)

# Function to calculate total within-cluster sum of square
wss <- function(k){
  kmeans(cluster_df,k,nstart=25)$tot.withinss
}

# Compute and plot wss for k=1 to k=15
k.values <- 2:12
# Extract wss for 2-15 clusters
wss_values <- map_dbl(k.values, wss)

# Plot elbow curve
plot(k.values, wss_values,
     type="b", pch=19, frame=FALSE,
     xlab = "Number of Clusters K",
     ylab = "Total within-clusters sum of squares")
```



```
## The results suggest that 6 is the optimal number of clusters as it appears to  
## be the bend of the elbow curve.
```