Clustering_VasanthakumarKalaikkovan

Vasanthakumar Kalaikkovan

29/05/2021

Clustering

In this problem, you will use the k-means clustering algorithm to look for patterns in an unlabeled dataset. The dataset for this problem is found at data/clustering-data.csv.

```
library(stats)
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.0.5
```

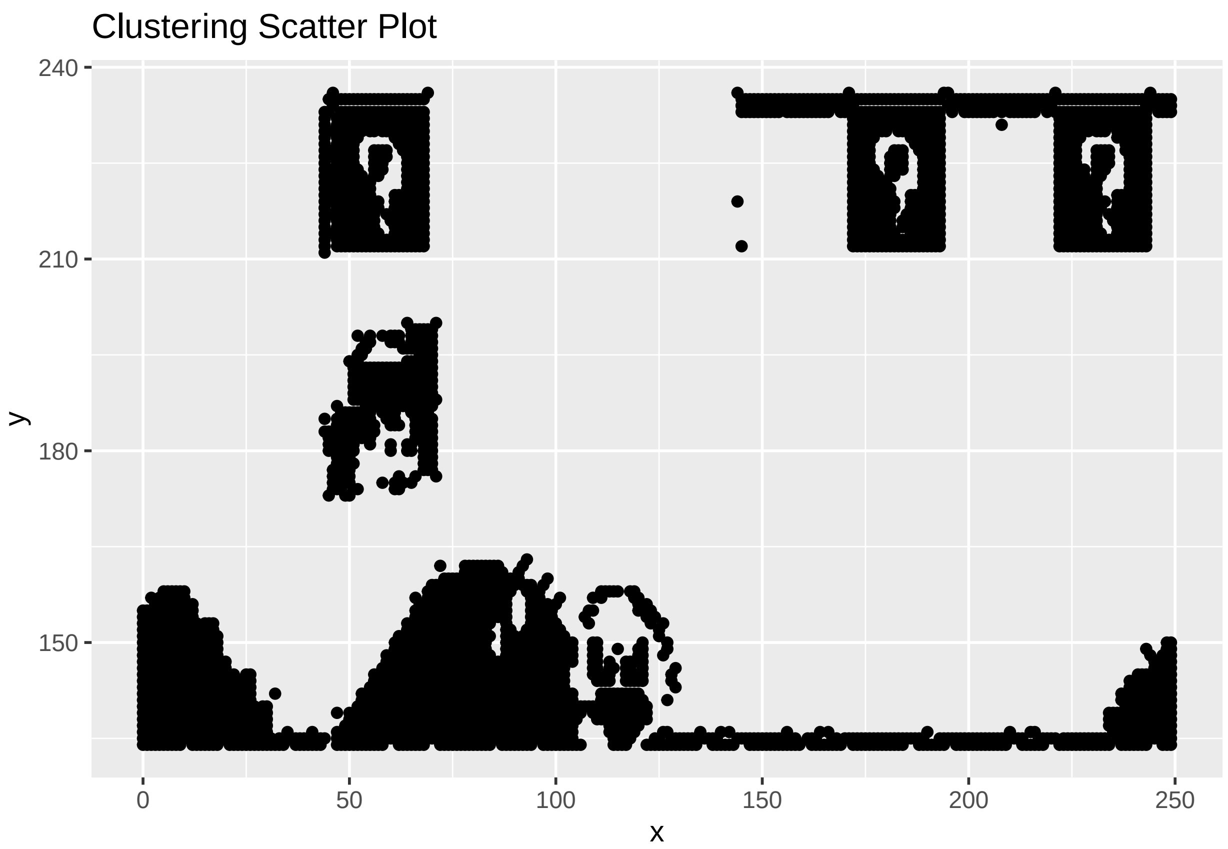
```
getwd()
```

```
## [1] "E:/Repos/StatisticsR/DSC520-Statistics/week11"
```

```
setwd("E://Repos/StatisticsR/DSC520-Statistics/week11")

cluster_df <- read.csv("clustering-data.csv")
```

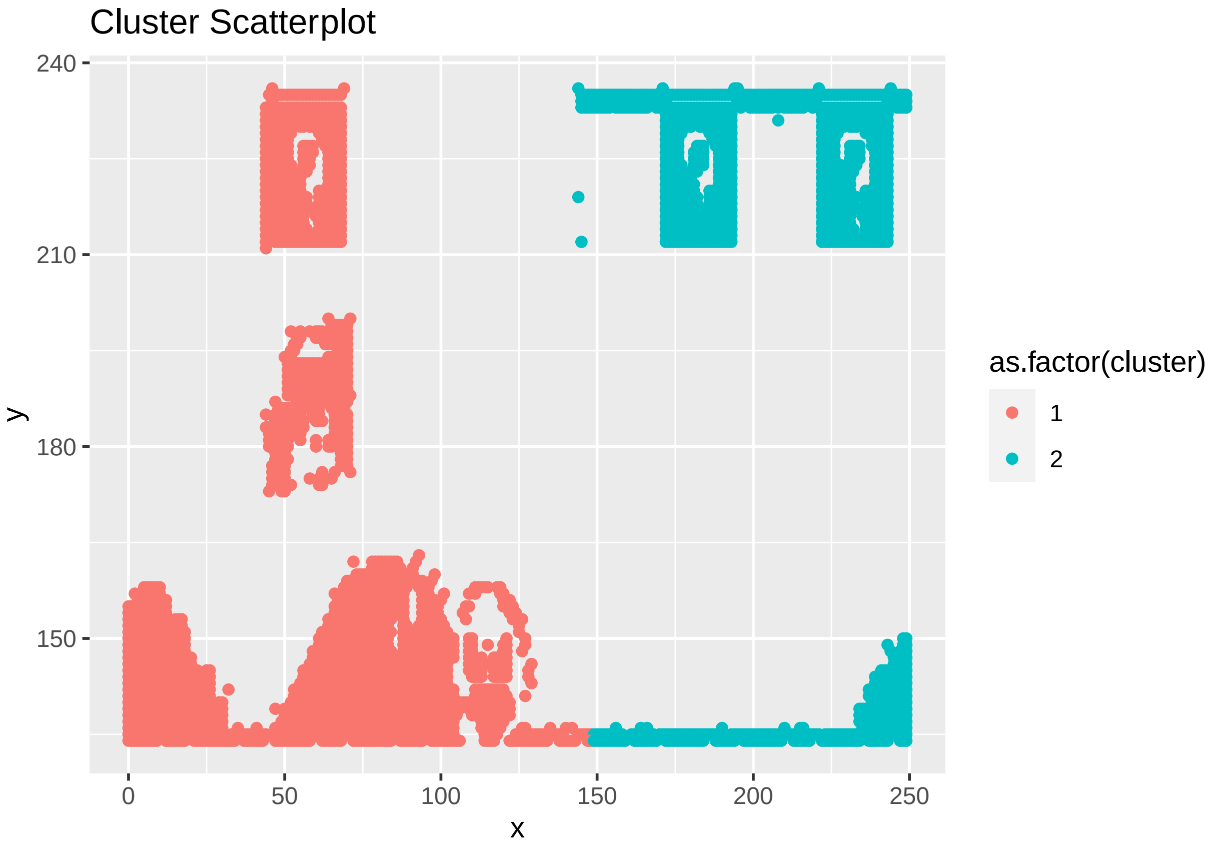
Plot the dataset using a scatter plot.

```
ggplot(cluster_df,aes(x=x,y=y))+geom_point()+labs(title="Clustering Scatter Plot")
```

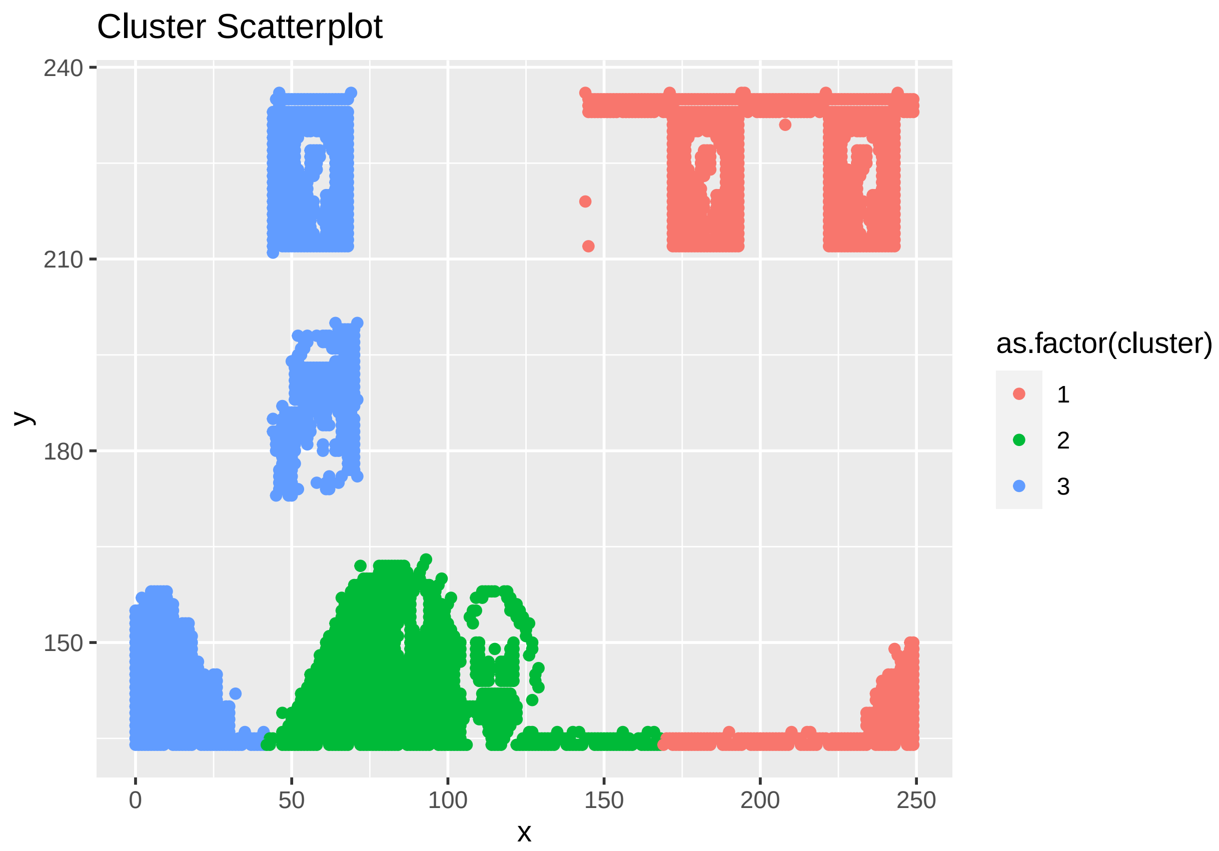


Fit the dataset using the k-means algorithm from k=2 to k=12. Create a scatter plot of the resultant clusters for each value of k.

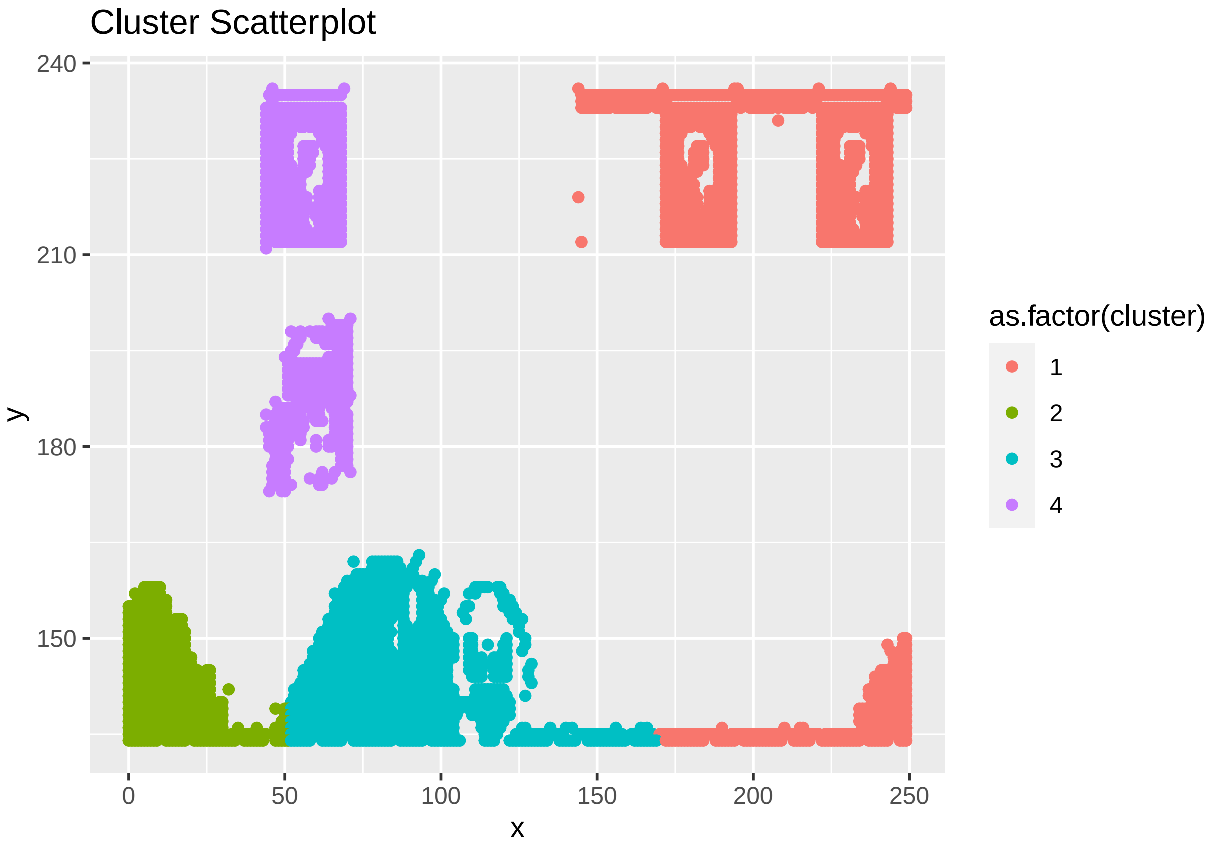
```
set.seed(2345)
clusters <-NULL
avg_dist <- NULL

for(i in 2:12)
{
  clusters_kMeans <- kmeans(cluster_df,i)
  clusters[i] <- as.data.frame(clusters_kMeans[["cluster"]])
  cluster_df["cluster"] <- clusters[i]

  avg_dist[i] <- sum(clusters_kMeans[["withinss"]]/clusters_kMeans[["size"]])

  print(ggplot(cluster_df,aes(x=x,y=y,color=as.factor(cluster)))+geom_point()+labs(title = "Cluster Sca
  print(avg_dist[i])
}
```


Cluster Scatterplot
240
210
180
150
y
0
50
100
150
200
250
x
as.factor(cluster)
● 1
● 2

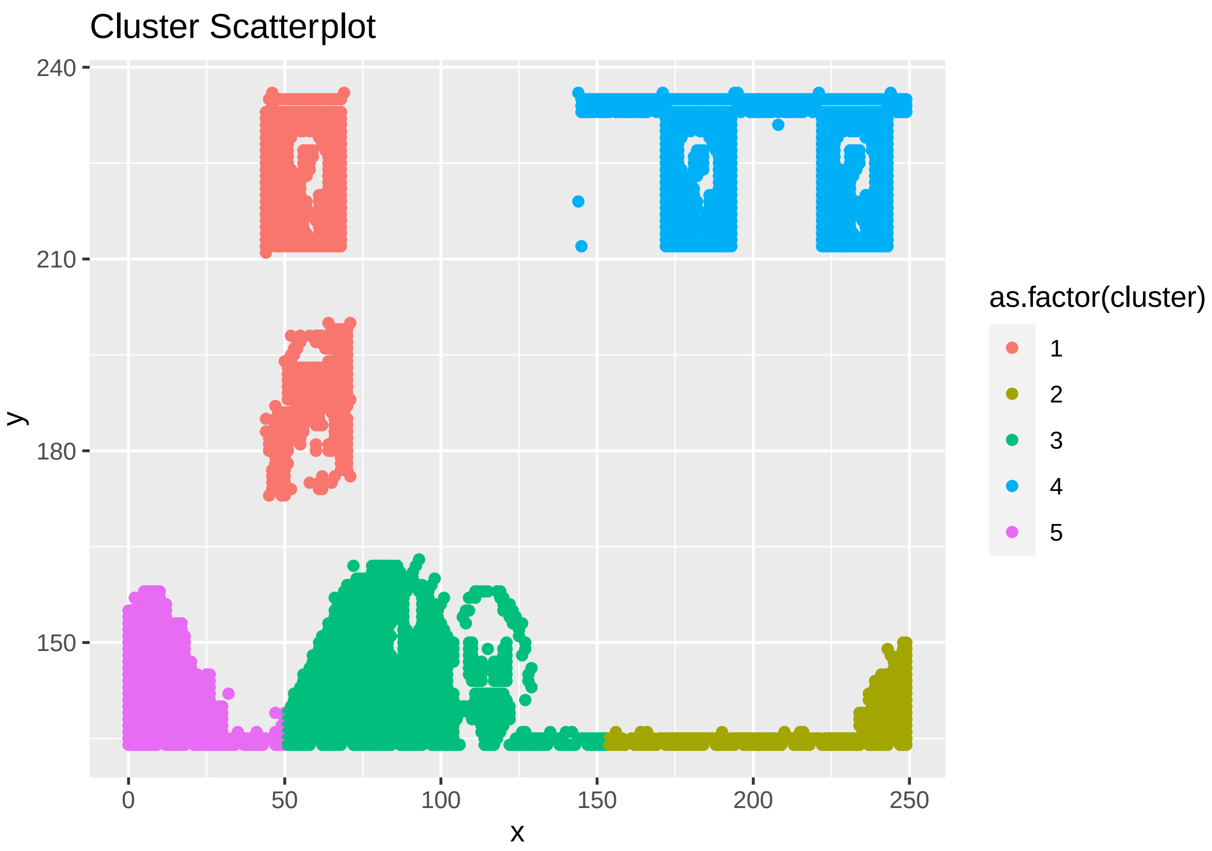
```
## [1] 4322.313
```


Cluster Scatterplot
240
210
180
150
y
0
50
100
150
200
250
x
as.factor(cluster)
● 1
● 2
● 3

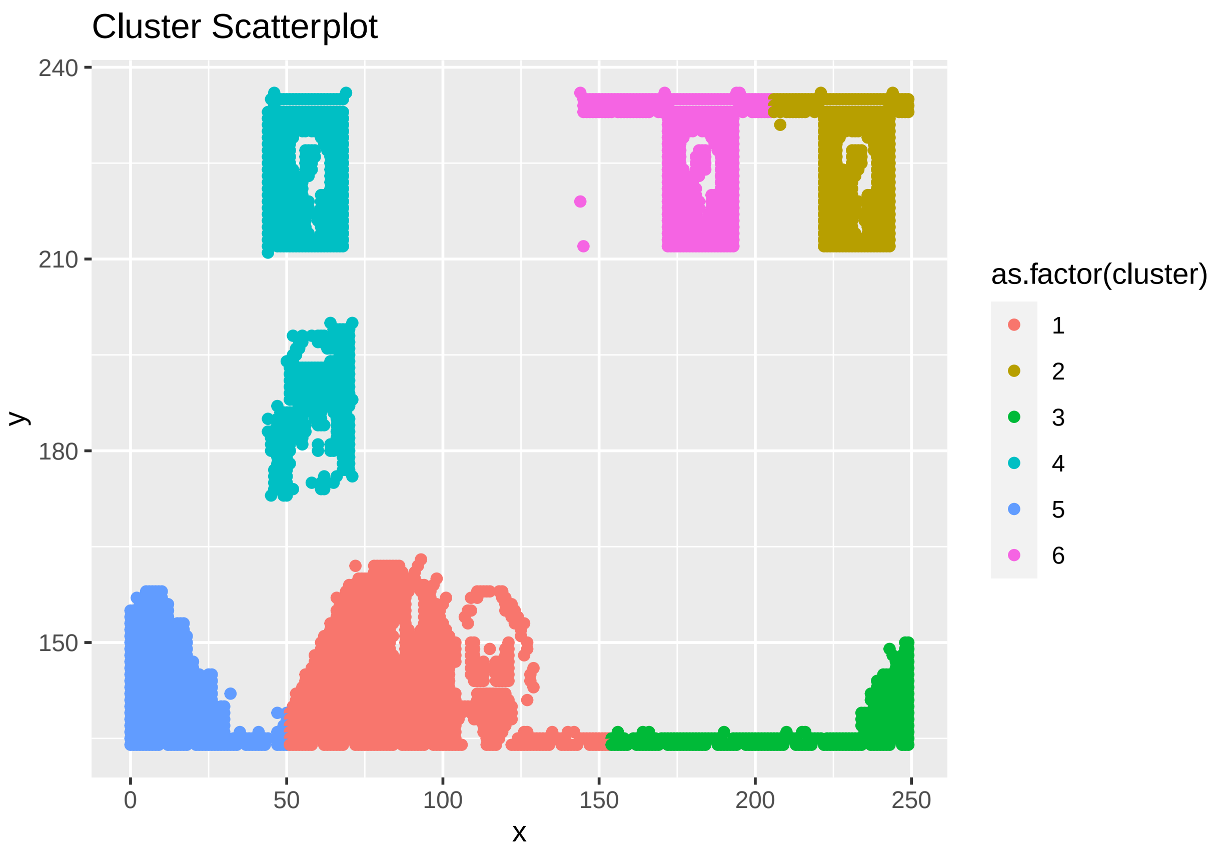
```
## [1] 4562.045
```


Cluster Scatterplot
240
210
180
150
y
0
50
100
150
200
250
x
as.factor(cluster)
● 1
● 2
● 3
● 4

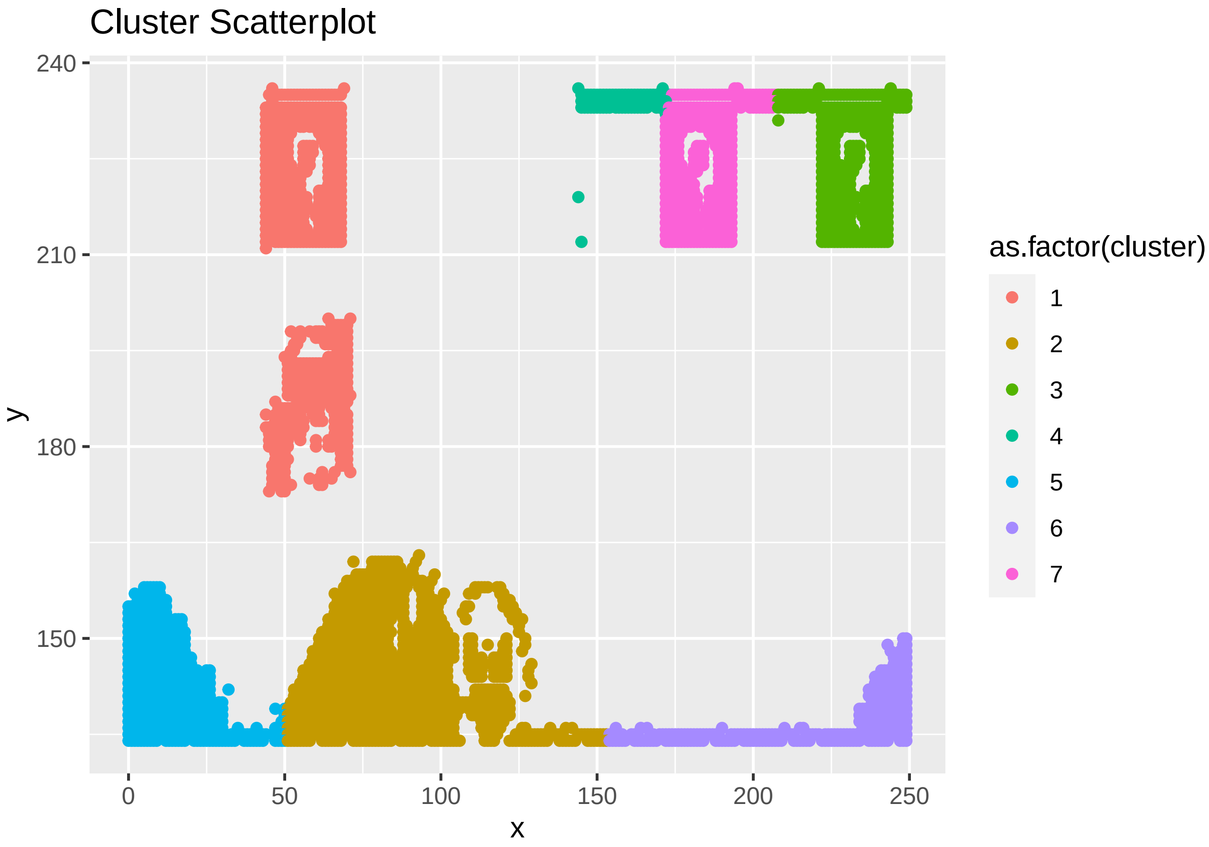
```
## [1] 3366.235
```


Cluster Scatterplot
240
210
180
150
y
0
50
100
150
200
250
x
as.factor(cluster)
1
2
3
4
5

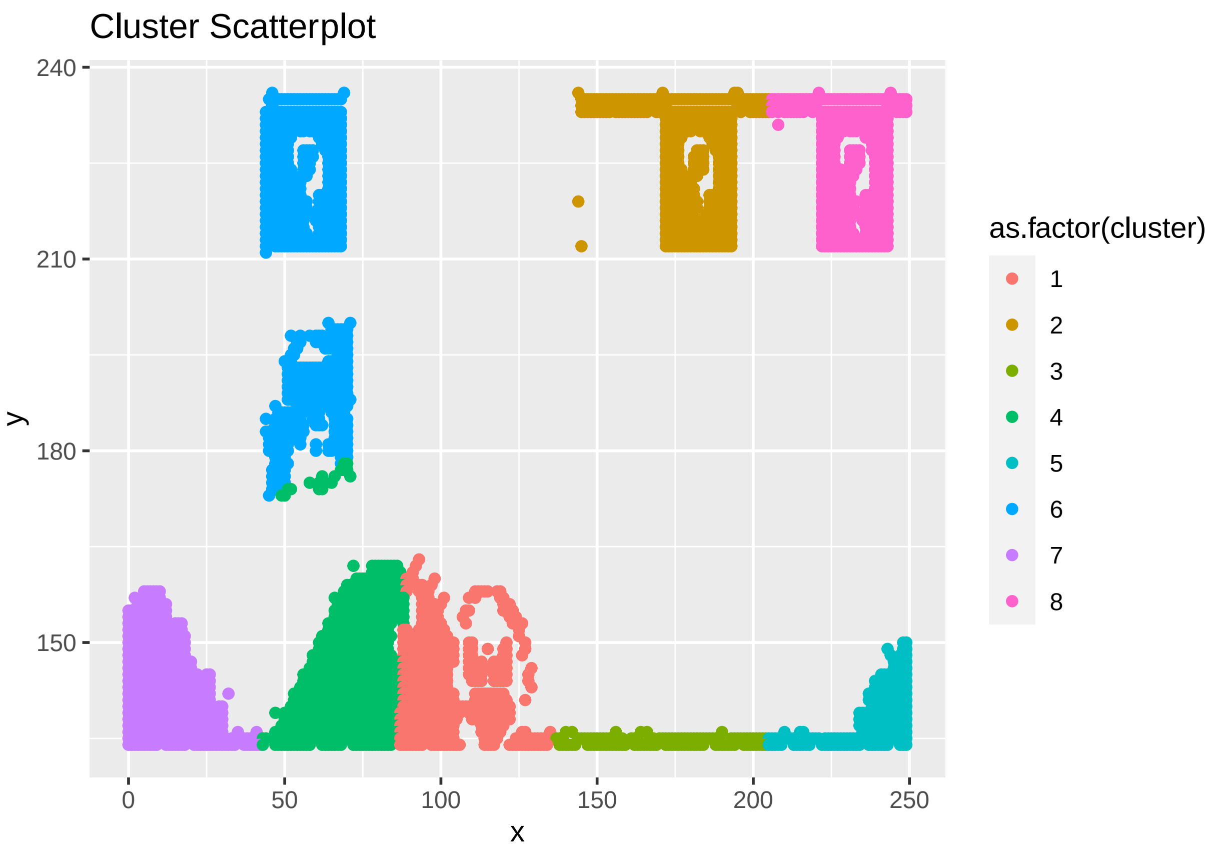
```
## [1] 2765.524
```


Cluster Scatterplot
240
210
180
150
y
0
50
100
150
200
250
x
as.factor(cluster)
● 1
● 2
● 3
● 4
● 5
● 6

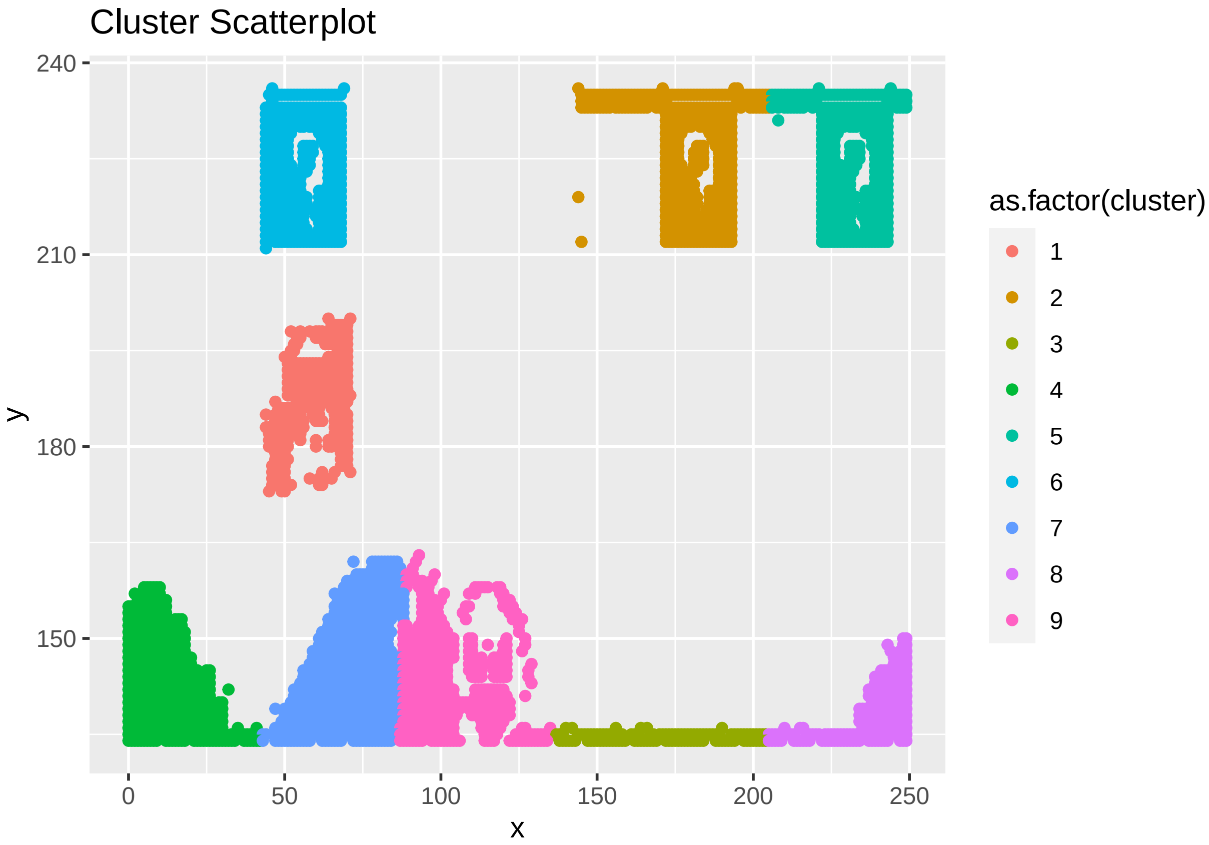
```
## [1] 2280.247
```


Cluster Scatterplot
240
210
180
150
y
0
50
100
150
200
250
x
as.factor(cluster)
● 1
● 2
● 3
● 4
● 5
● 6
● 7

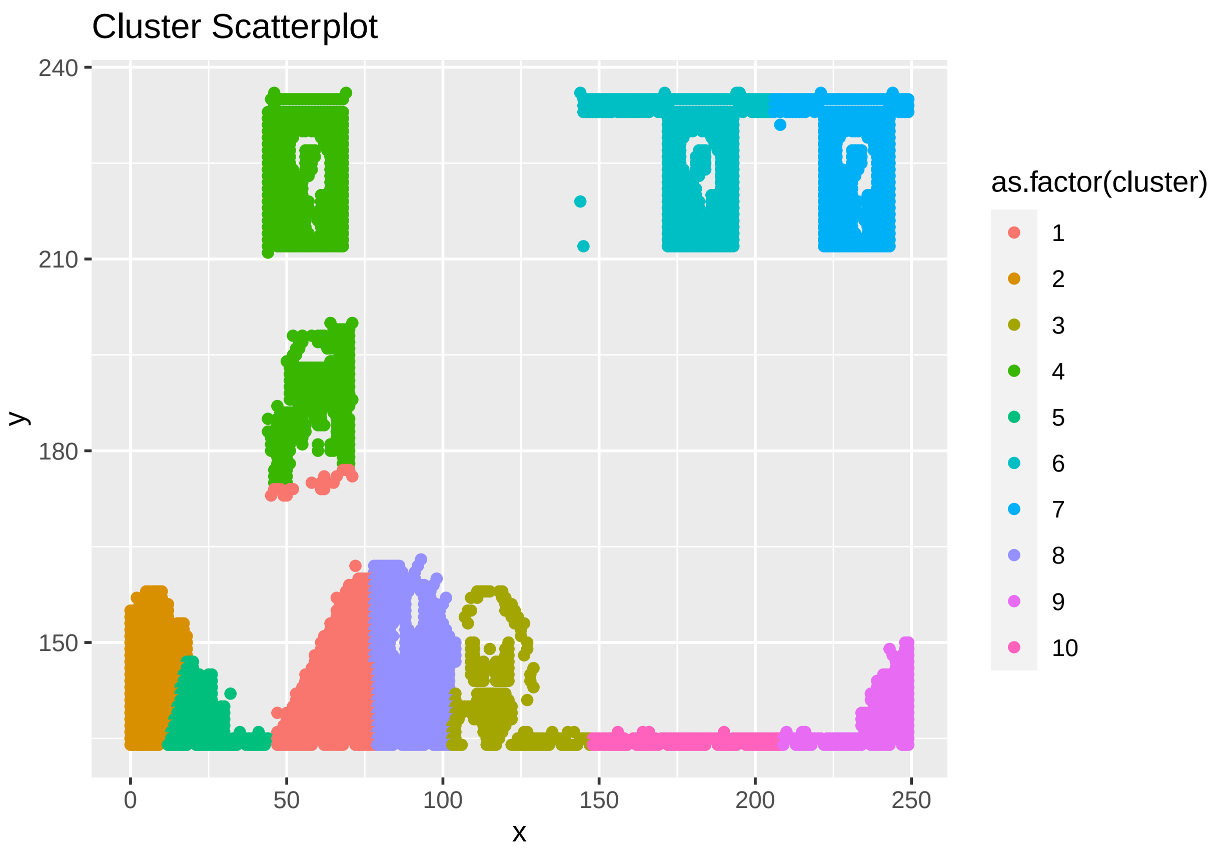
```
## [1] 2261.702
```


Cluster Scatterplot
240
210
180
150
y
0
50
100
150
200
250
x
as.factor(cluster)
1
2
3
4
5
6
7
8

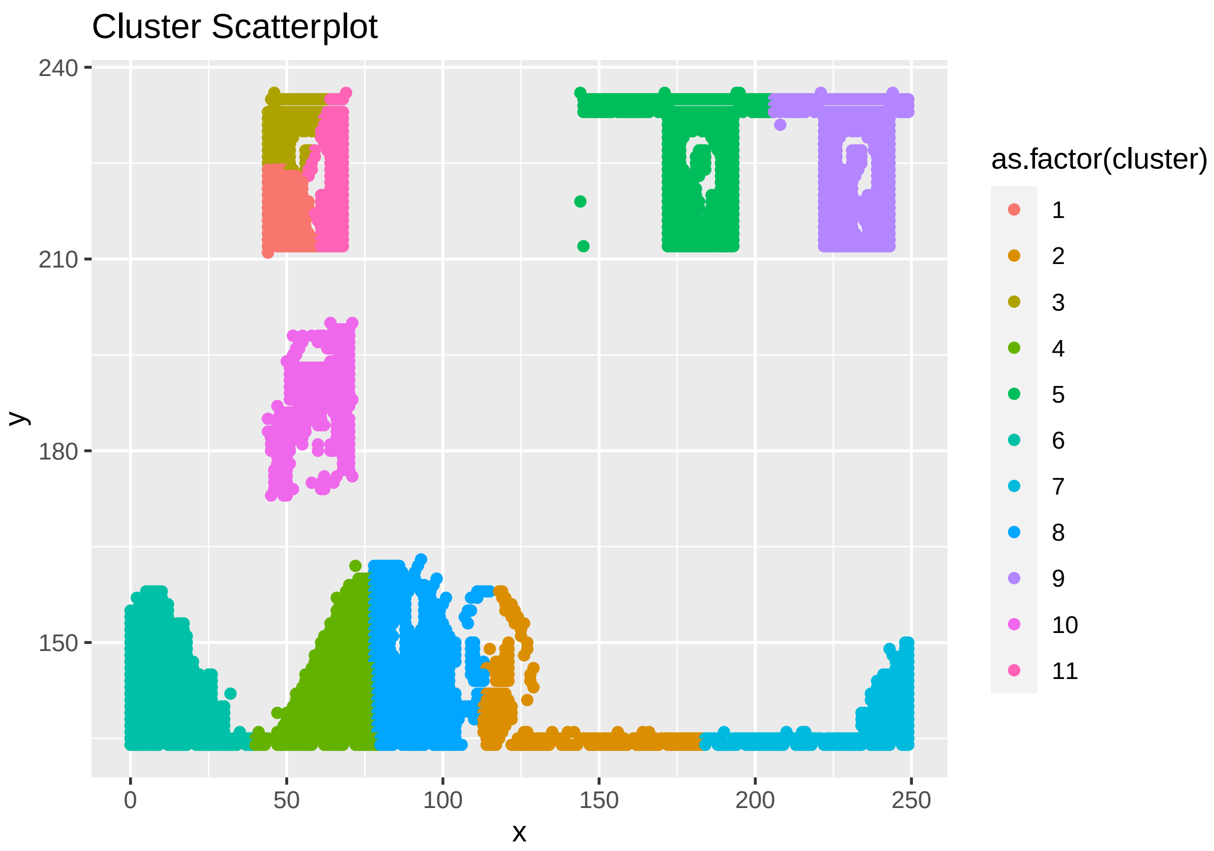
```
## [1] 1806.729
```


Cluster Scatterplot
240
210
180
150
y
0
50
100
150
200
250
x
as.factor(cluster)
1
2
3
4
5
6
7
8
9

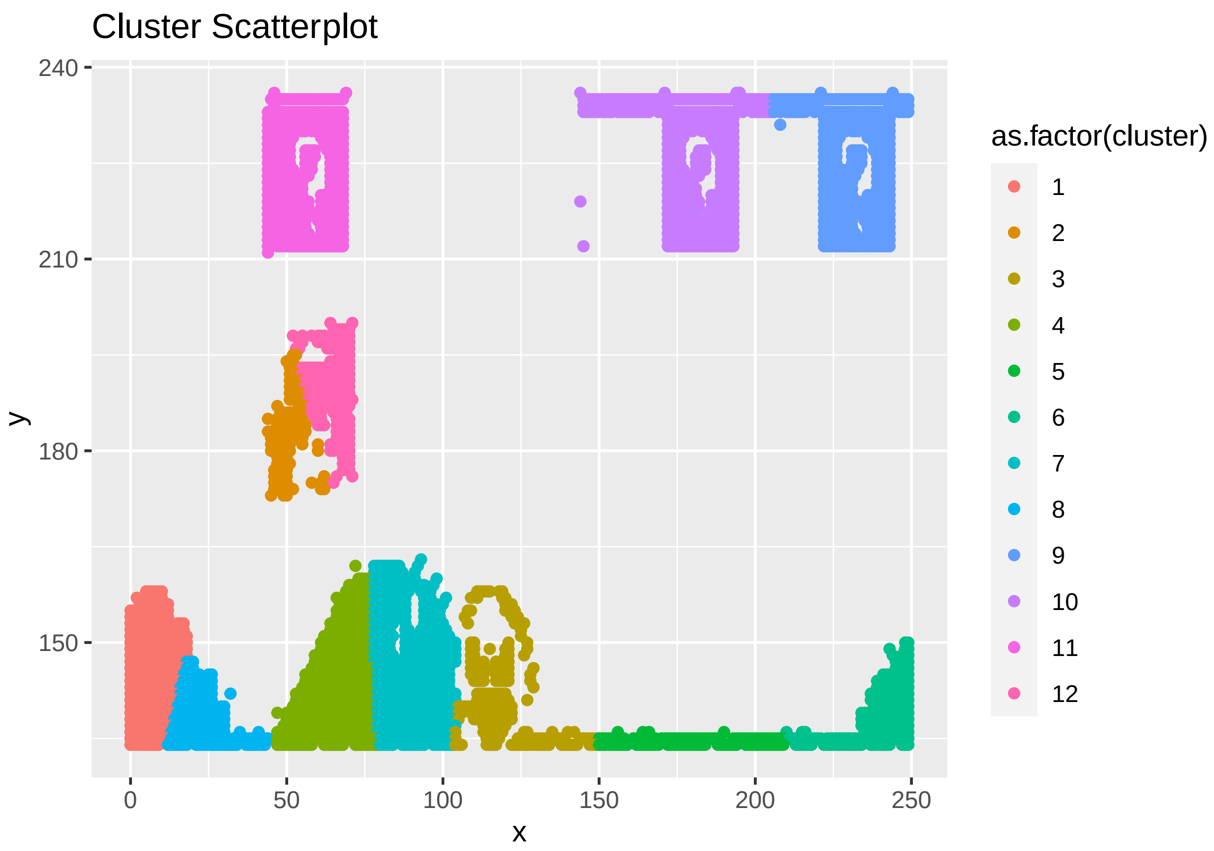
```
## [1] 1584.719
```


Cluster Scatterplot
240
210
180
150
y
0
50
100
150
200
250
x
as.factor(cluster)
● 1
● 2
● 3
● 4
● 5
● 6
● 7
● 8
● 9
● 10

```
## [1] 1743.035
```


Cluster Scatterplot
240
210
180
150
y
0
50
100
150
200
250
x
as.factor(cluster)
● 1
● 2
● 3
● 4
● 5
● 6
● 7
● 8
● 9
● 10
● 11

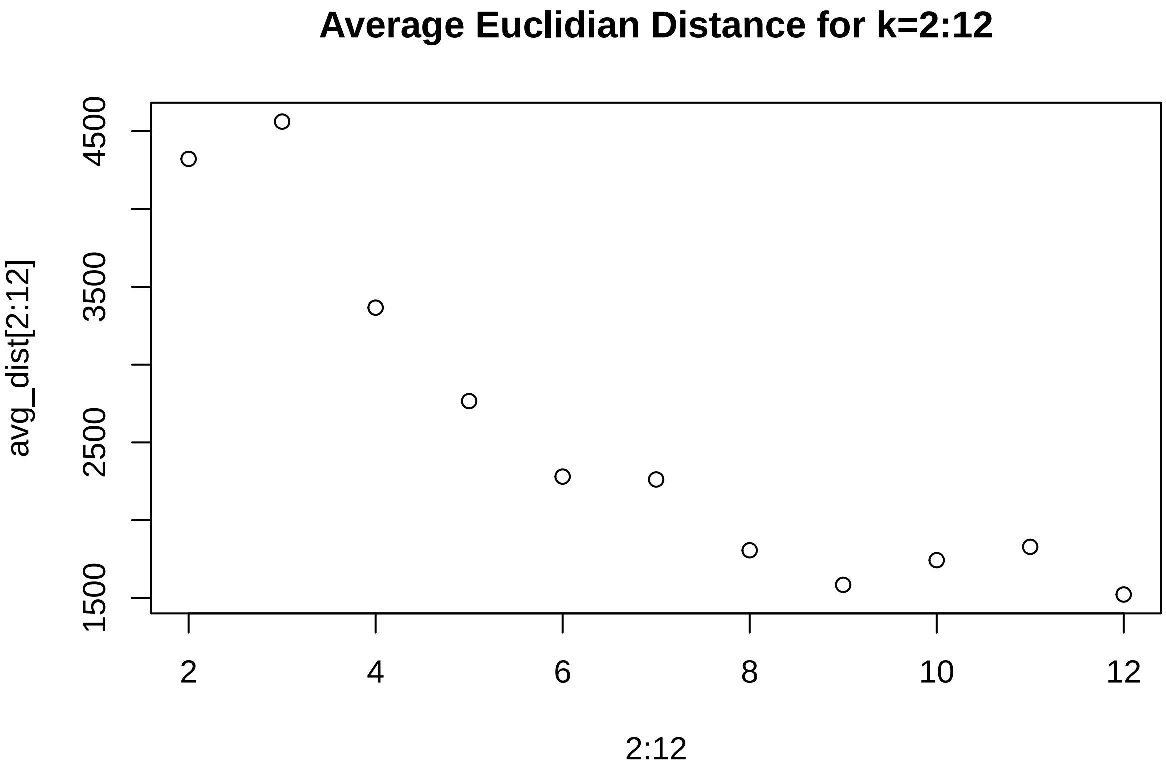
```
## [1] 1828.966
```



```
## [1] 1522.473
```

Calculate this average distance from the center of each cluster for each value of k and plot it as a line chart where k is the x-axis and the average distance is the y-axis. One way of determining the "right" number of clusters is to look at the graph of k versus average distance and finding the "elbow point". Looking at the graph you generated in the previous example, what is the elbow point for this dataset?

```
plot(2:12,avg_dist[2:12],main="Average Euclidian Distance for k=2:12")
```


Average Euclidian Distance for k=2:12
avg_dist[2:12]
4500
3500
2500
1500
2
4
6
8
10
12
2:12

Elbow, k=9