Kmeans

2024-06-21

Load library

```
library(readr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(tidyr)
library(cluster)    # clustering algorithms
library(factoextra)
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WB
a
```

```
library(purrr)
library(plotly)
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following object is masked from 'package:graphics':
##
##     layout
```

Import dataset

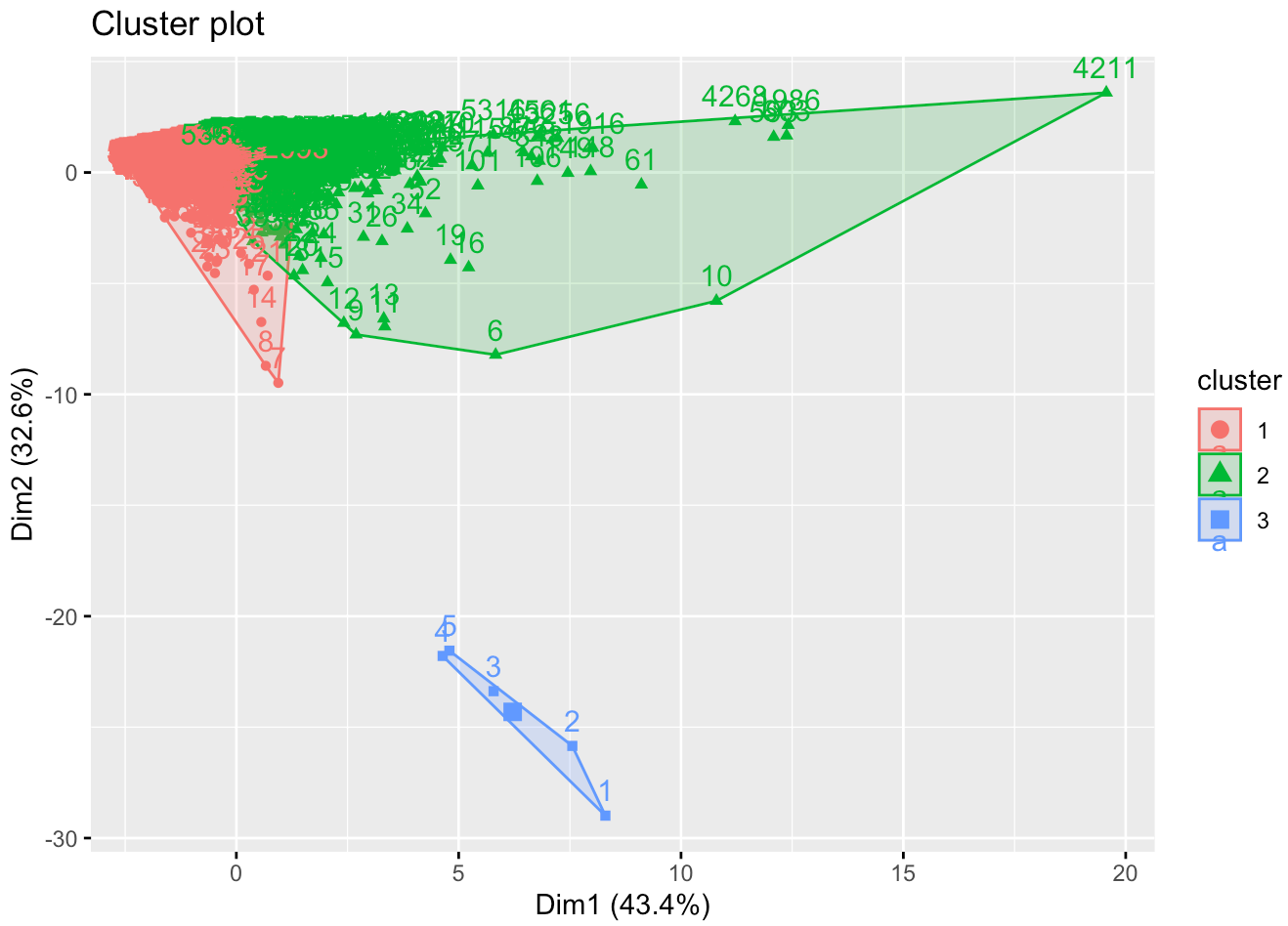
```
rank <- read.csv("rank.csv")
# Examine data
head(rank)
```

	Customer.ID	last_date	recency	frequency	average_invoice_value	rec_rank	rank_freq	ra
	<int>	<chr>	<int>	<int>	<dbl>	<int>	<int>	
1	15749	2011-04-18	235	3	14844.767	3400	4211	
2	15098	2011-06-10	182	3	13305.500	3148	1986	
3	13687	2010-09-27	438	1	11880.840	4500	603	
4	12918	2010-03-23	626	1	10953.500	5087	593	
5	18052	2010-05-24	564	1	10877.180	4904	4268	
6	17450	2011-12-01	8	51	4799.691	485	1916	

6 rows | 1-9 of 13 columns

One thing to notice here, the magnitude of difference between the average invoice and the recency/frequency is too high. They should be in the same scale. So we need to scale the feature and create the segmentation.

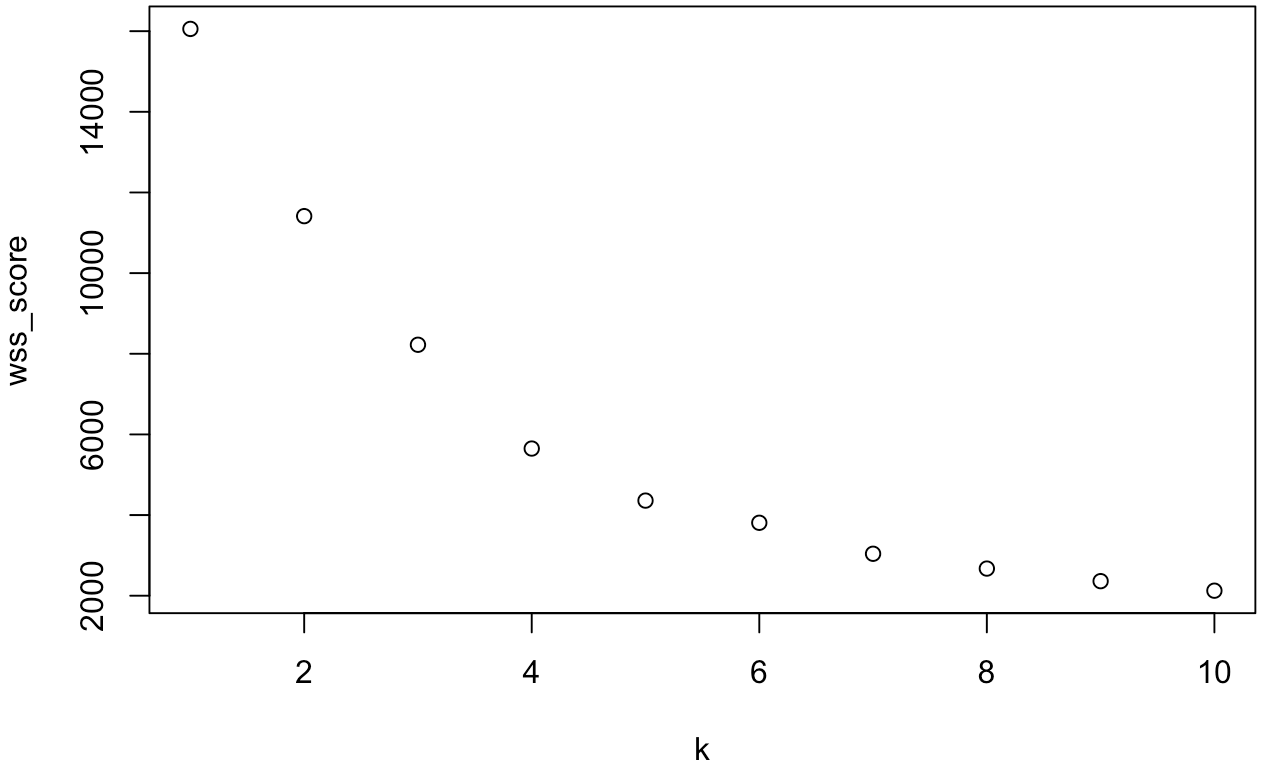
```
a <- kmeans(scale(rank[, 3:5]), #recency, frequency, avg invoice
                  centers = 3,
                  iter.max = 18, # 18 iteration till conversion
                  nstart = 10)

## a$cluster

# Visualize
fviz_cluster(a, rank[,3:5])
```



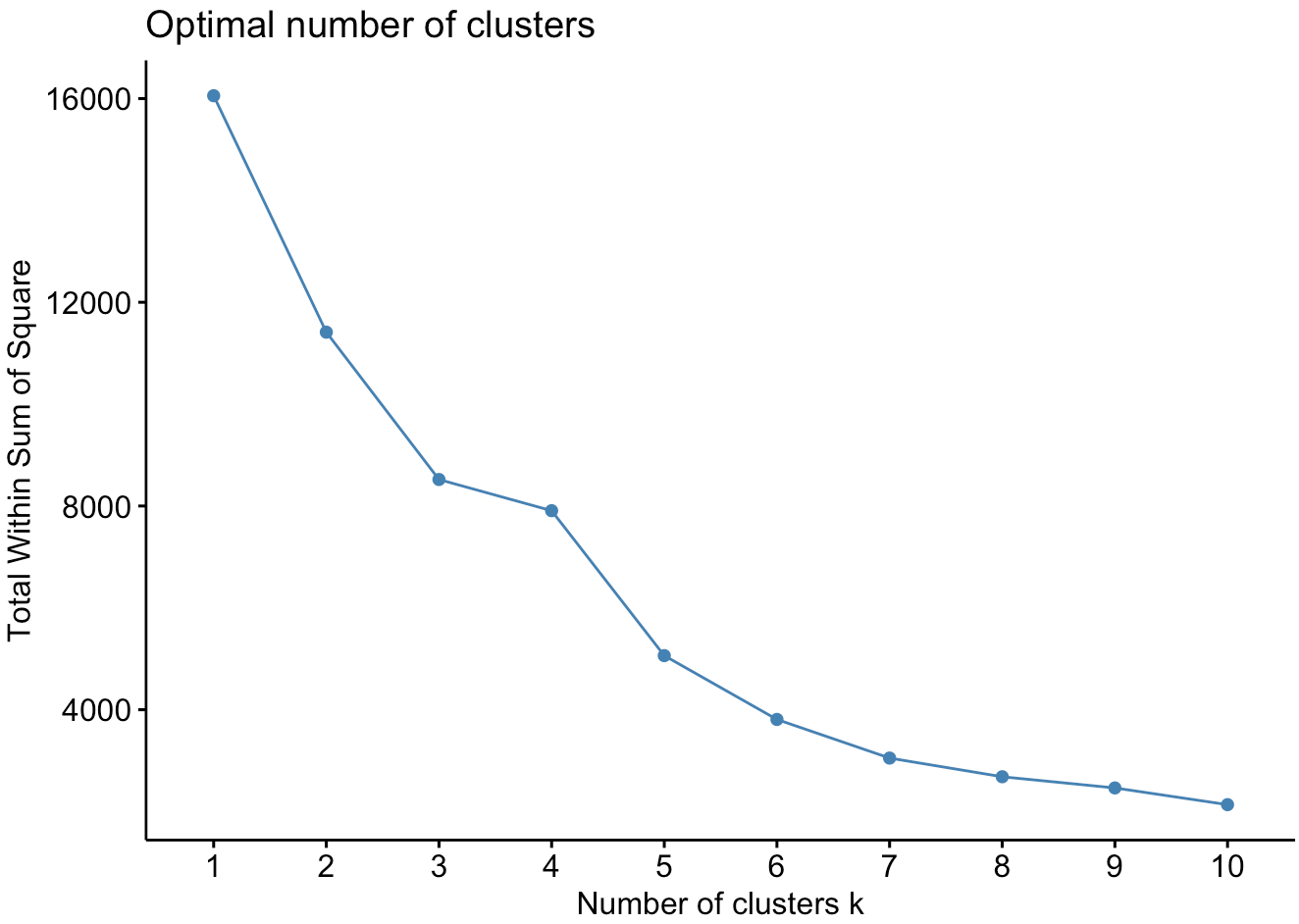

```
# Examine the total within this cluster
a$tot.withinss
```

```
## [1] 8223.494
```

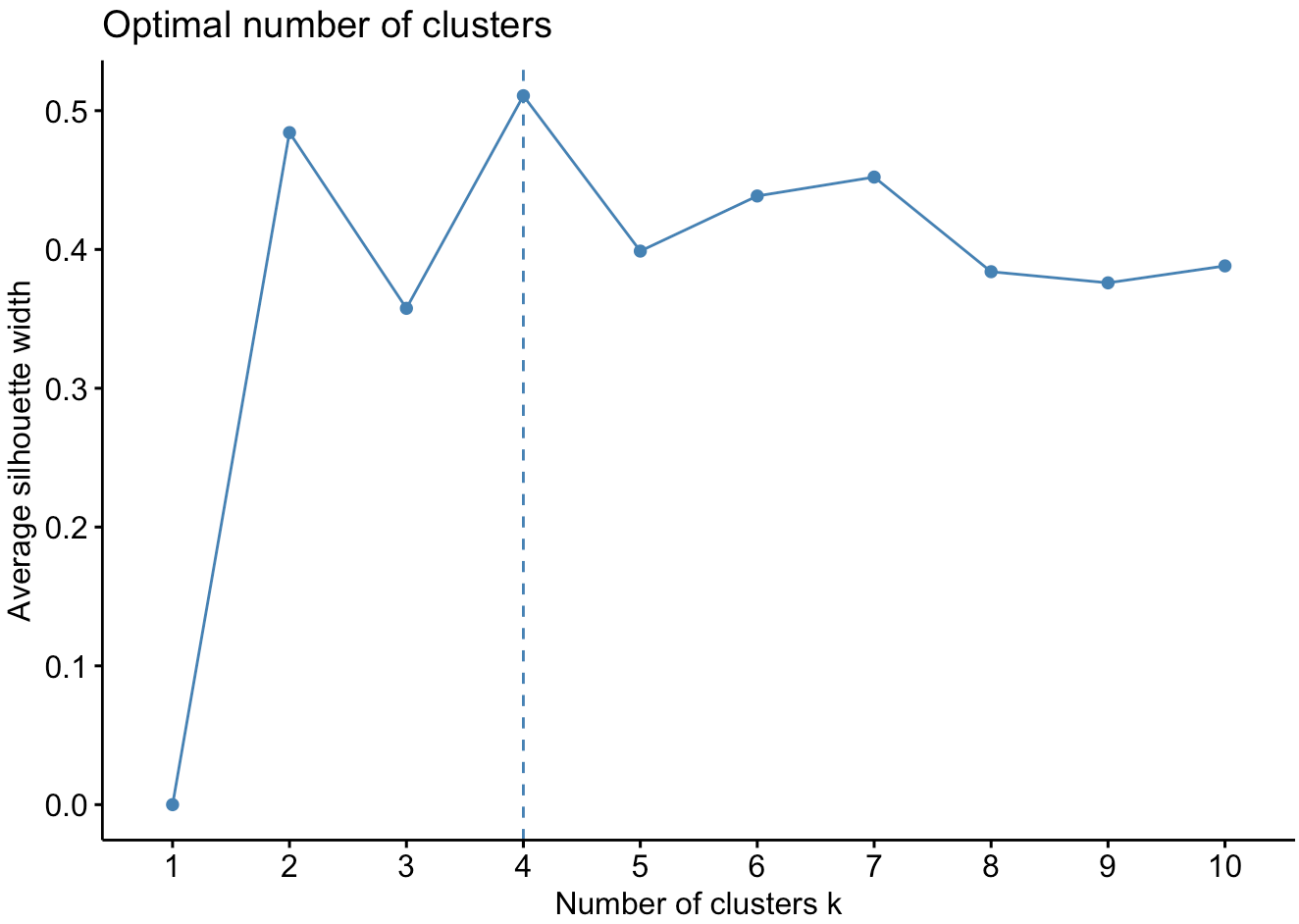
```
# Create functions to test different clusters
wss <- function(k){
  kmeans(scale(rank[,3:5]), , centers = k, iter.max = 18, nstart =10)$tot.withinss
}

# Let try 10 k
k<- seq(1:10)

# Apply mapping function
wss_score<-map_dbl(k, wss)

# Plot to examine
plot(k,wss_score)
```



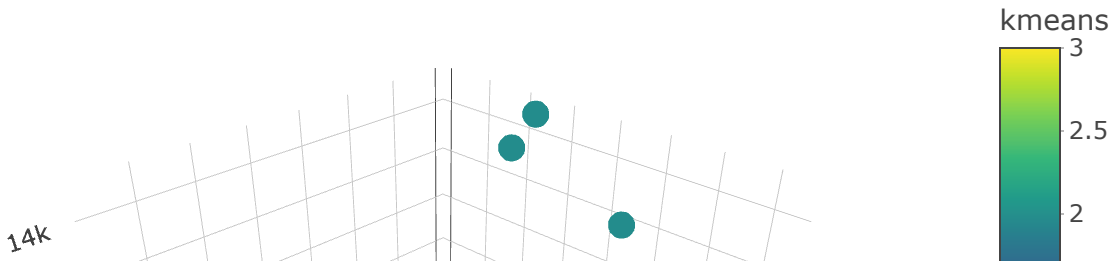

```
#Verify the cut-off point
fviz_nbclust(scale(rank[,3:5]), kmeans, method = "wss")
```

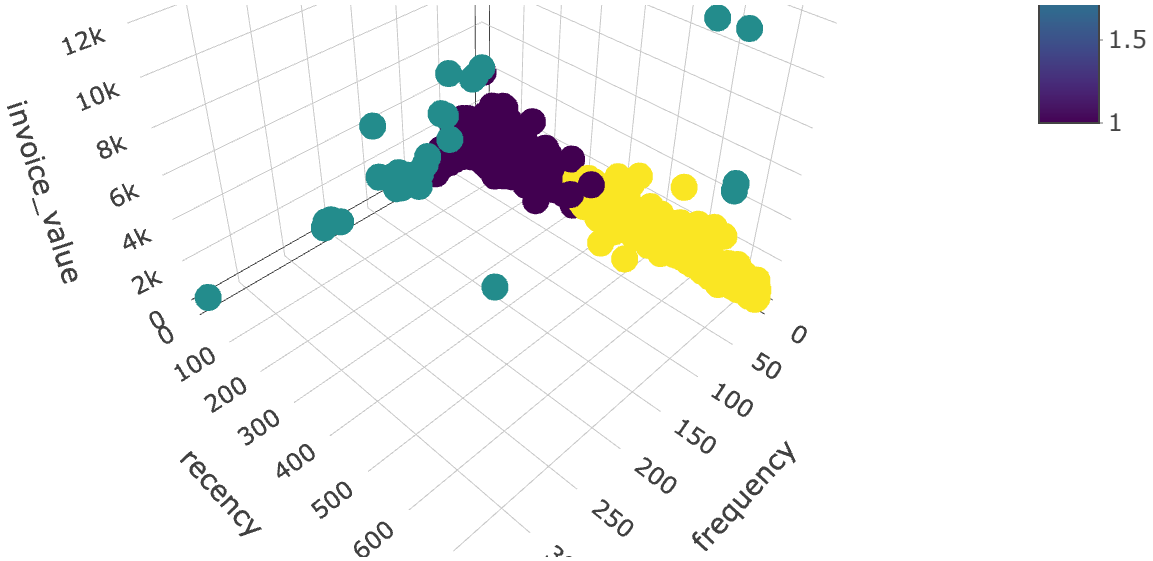



```
fviz_nbclust(scale(rank[,3:5]), kmeans, method = "silhouette")
```




```
# Optimum clusters
b <- kmeans(scale(rank[, 3:5]),
              centers = 3,
              iter.max = 18,
              nstart = 10)

## 3 dimensional scatter plot
rank$kmeans<- b$cluster

fig <- plot_ly(rank, x = ~frequency, y = ~recency, z = ~average_invoice_value,
               color = ~kmeans )
fig <- fig %>% add_markers()
fig <- fig %>% layout(scene = list(xaxis = list(title = 'frequency'),
                                   yaxis = list(title = 'recency'),
                                   zaxis = list(title = 'invoice_value')))
fig
```





As we can see, we have some high invoice value, they have recency and low frequence. We can conclude that they are 1 time buyer. The difference about the rest 3 group are the recency.