

ML 1000 Assignment 2

by Anupama r.k, Queenie Tsang, Crystal (Yunan) Zhu

21/02/2021

To do list:

- Add Pie charts! - by sub_category, region # (done)
- Create a Month variable - to see the change of sales/profits by month?
- bar charts of profits/sales by region #(done)
- Output the characteristics of the orders with the highest and lowest profits/sales - e.g. what made the order? when? bought what product? in which city/state/region? Any discount?
- relationship between discount & sales, discount & profits, sales & profits, and the role of region?
- from someone's analysis - there is no significant change between the four discount categories when it comes to Sales
- sales/profits by month, rather than by date? color by region?

Abstract

Anomaly detection or Outlier detection identifies data points, events or observations that deviate from dataset's normal behavior. Anomalous data indicate critical incidents or potential opportunities. In order to take advantage of opportunities or fix costly problems anomaly detection has to be done in real time. Unsupervised machine learning models can be used to automate anomaly detection. Unsupervised anomaly detection algorithms scores data based on intrinsic properties of the dataset. Distances and densities are used to give an estimation what is normal and what is an outlier. Anomaly detection monitor is a tool developed for an online retailer to check product quality issues like profit opportunities and sales glitches. The application is built using R and Shinyapp following CRISP-DM framework.

Business Case

Objective

Detect point anomalies from superstore dataset using K-NN and clustering methods.

Data Understanding

US Superstore dataset is sourced from US uperstore dataset . The dataset have online orders for Superstores in U.S. from 2014-2018. Tableau community is the owner of the dataset. The dataset has 9994 records and 21 attributes.

Import data

```
superstore<- read_excel("US_Superstore_data.xls")

## Warning in read_fun(path = enc2native(normalizePath(path)), sheet_i = sheet, :
## Coercing text to numeric in L2236 / R2236C12: '05408'

## Warning in read_fun(path = enc2native(normalizePath(path)), sheet_i = sheet, :
## Coercing text to numeric in L5276 / R5276C12: '05408'

## Warning in read_fun(path = enc2native(normalizePath(path)), sheet_i = sheet, :
## Coercing text to numeric in L8800 / R8800C12: '05408'

## Warning in read_fun(path = enc2native(normalizePath(path)), sheet_i = sheet, :
## Coercing text to numeric in L9148 / R9148C12: '05408'

## Warning in read_fun(path = enc2native(normalizePath(path)), sheet_i = sheet, :
## Coercing text to numeric in L9149 / R9149C12: '05408'

## Warning in read_fun(path = enc2native(normalizePath(path)), sheet_i = sheet, :
## Coercing text to numeric in L9150 / R9150C12: '05408'

## Warning in read_fun(path = enc2native(normalizePath(path)), sheet_i = sheet, :
## Coercing text to numeric in L9388 / R9388C12: '05408'

## Warning in read_fun(path = enc2native(normalizePath(path)), sheet_i = sheet, :
## Coercing text to numeric in L9389 / R9389C12: '05408'

## Warning in read_fun(path = enc2native(normalizePath(path)), sheet_i = sheet, :
## Coercing text to numeric in L9390 / R9390C12: '05408'

## Warning in read_fun(path = enc2native(normalizePath(path)), sheet_i = sheet, :
## Coercing text to numeric in L9391 / R9391C12: '05408'

## Warning in read_fun(path = enc2native(normalizePath(path)), sheet_i = sheet, :
## Coercing text to numeric in L9743 / R9743C12: '05408'

data_superstore
```

Table 1: Dataset description

Attribute	Data Type	Description
Row ID	numeric	row number
Order ID	character	unique order number
Order Date	numeric	order placed date
Ship Date	numeric	order shipping date
Ship Mode	character	shipping mode of order
Customer ID	character	unique customer id for order

Attribute	Data Type	Description
Customer Name	character	name of customer
Segment	character	section of product
Country	character	country based on order
City	character	city based on order
State	character	state based on order
Postal Code	numeric	pin code
Region	character	region based on order
Product ID	character	product id of product
Category	character	category of product
Sub-Category	character	sub-category of product
Product Name	character	name of product
Sales	numeric	selling price of product
Quantity	numeric	order quantity
Discount	numeric	discount on product
Profit	numeric	profit from product

```

## [1] "i..Row.ID-0 missing values"      "Order.ID-0 missing values"
## [3] "Order.Date-0 missing values"     "Ship.Date-0 missing values"
## [5] "Ship.Mode-0 missing values"       "Customer.ID-0 missing values"
## [7] "Customer.Name-0 missing values"   "Segment-0 missing values"
## [9] "Country-0 missing values"         "City-0 missing values"
## [11] "State-0 missing values"           "Postal.Code-0 missing values"
## [13] "Region-0 missing values"          "Product.ID-0 missing values"
## [15] "Category-0 missing values"        "Sub.Category-0 missing values"
## [17] "Product.Name-0 missing values"    "Sales-0 missing values"
## [19] "Quantity-0 missing values"        "Discount-0 missing values"
## [21] "Profit-0 missing values"          "diff_in_days-0 missing values"

```

Get a general idea of the data set.

```
length(unique(data$Customer.ID))
```

```
## [1] 793
```

#793 unique customer IDs

```
length(unique(data$Customer.Name))
```

```
## [1] 793
```

#793 unique customer names - drop one of these two vars

```
length(unique(data$Order.Date))
```

```
## [1] 1237
```

#1237 unique order dates

```
length(unique(data$Ship.Date))
```

```

## [1] 1334

#1334 unique ship dates - more unique ship dates than order dates - orders made on the same day were shipped

length(unique(data$Segment))

## [1] 3

unique(data$Segment)

## [1] "Consumer"     "Corporate"    "Home Office"

#"Consumer"      "Corporate"    "Home Office"

unique(data$Country)

## [1] "United States"

#all are from US - could drop this variable due to no-variation introduced by it

length(unique(data$City))

## [1] 531

#531 different cities

length(unique(data$State))

## [1] 49

#49 states

length(unique(data$Postal.Code))

## [1] 631

#631 postal code - 793 unique customer IDs - some customers live very close!

unique(data$Region)

## [1] "South"      "West"       "Central"    "East"

#only 4 regions

unique(data$Category)

## [1] "Furniture"   "Office Supplies" "Technology"

```

```

#only 3 categories - "Furniture" "Office Supplies" "Technology"

length(unique(data$Sub.Category))

## [1] 17

unique(data$Sub.Category)

## [1] "Bookcases"    "Chairs"        "Labels"        "Tables"        "Storage"
## [6] "Furnishings"   "Art"           "Phones"        "Binders"       "Appliances"
## [11] "Paper"         "Accessories"   "Envelopes"     "Fasteners"     "Supplies"
## [16] "Machines"      "Copiers"

#17 sub-categories

length(unique(data$Product.Name))

## [1] 1850

#1850 product names
length(unique(data$Product.ID))

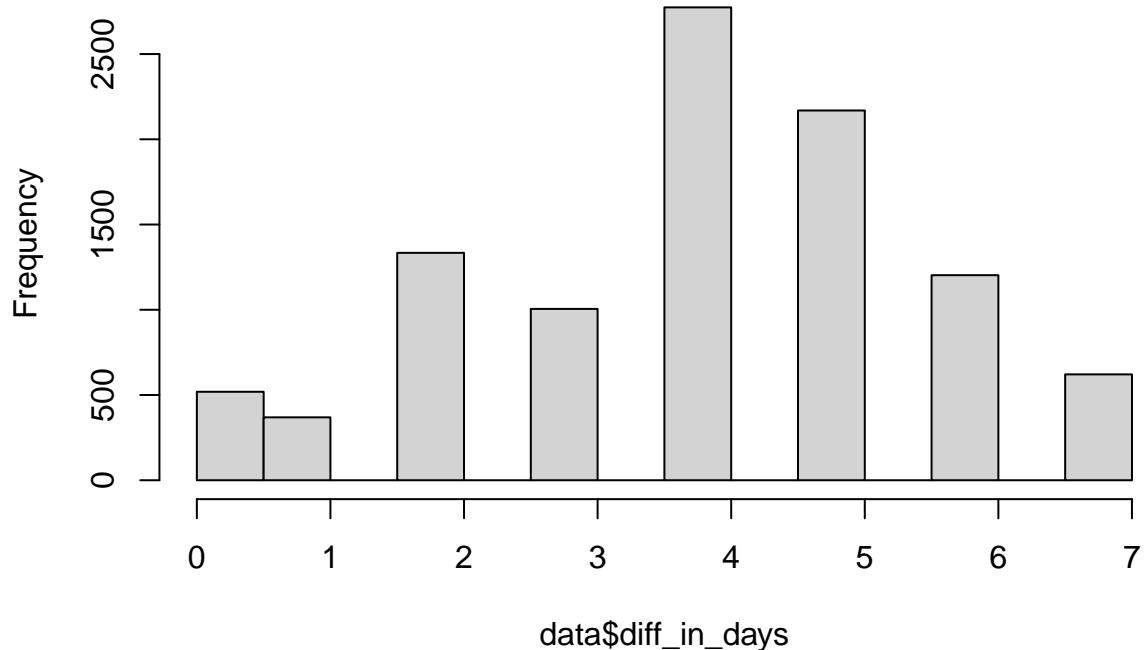
## [1] 1862

#1862 product IDs - potential redundant variables!

hist(data$diff_in_days)

```

Histogram of data\$diff_in_days

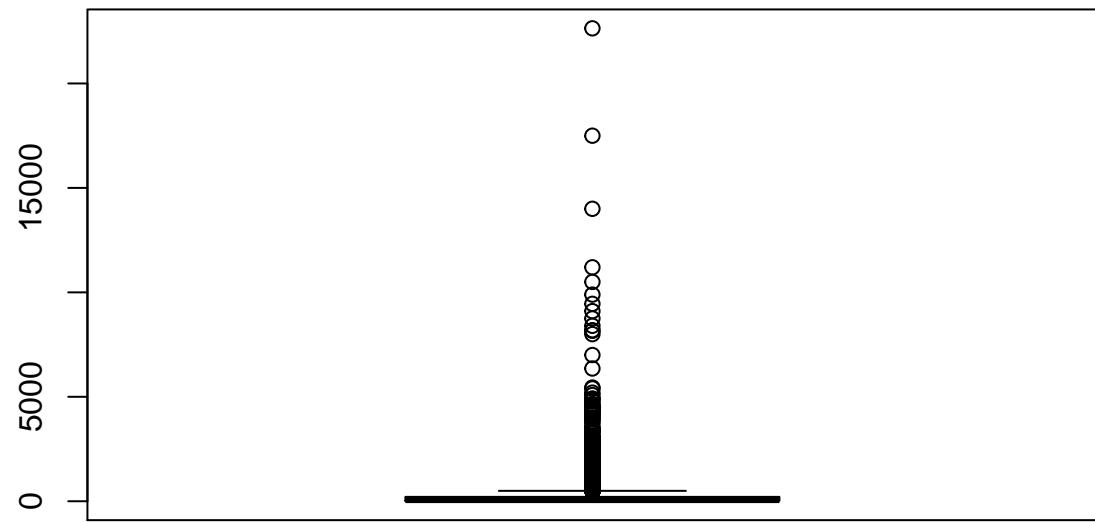


```
#The time difference between order date and ship date typically takes 4 days.
```

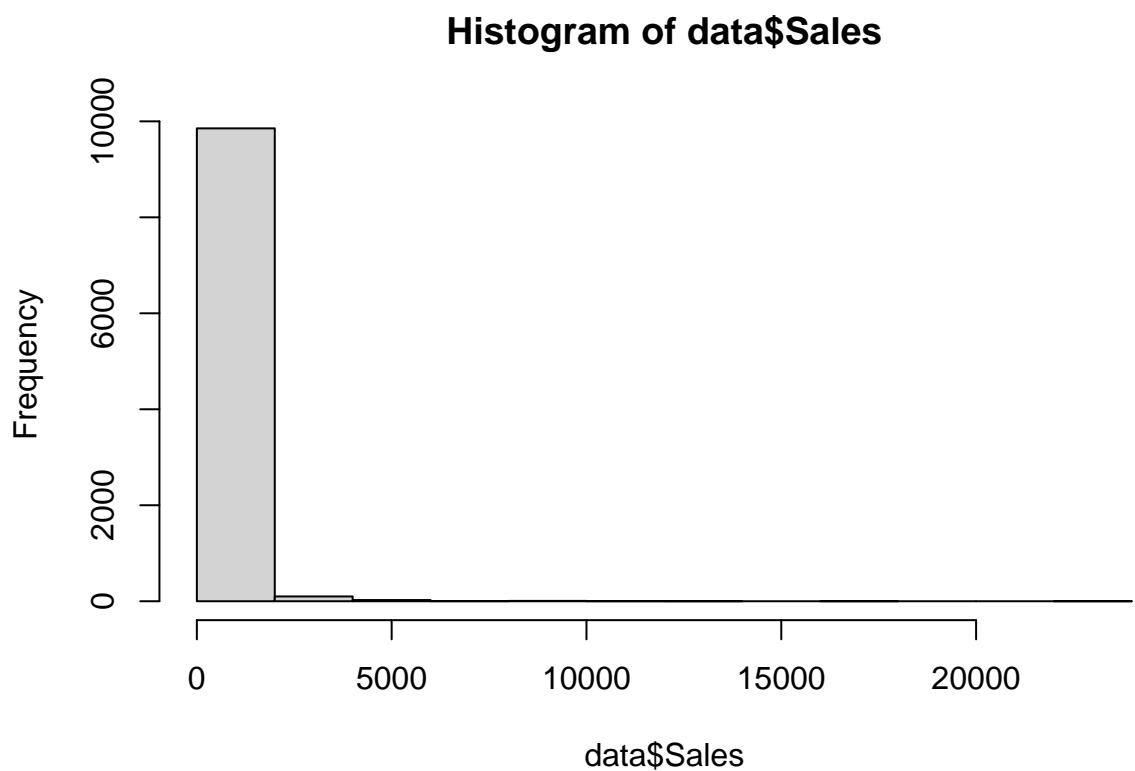
```
summary(data$Sales)
```

```
##      Min.    1st Qu.     Median      Mean    3rd Qu.      Max.
## 0.444    17.280    54.490   229.858  209.940  22638.480
```

```
boxplot(data$Sales)
```



```
hist(data$Sales)
```

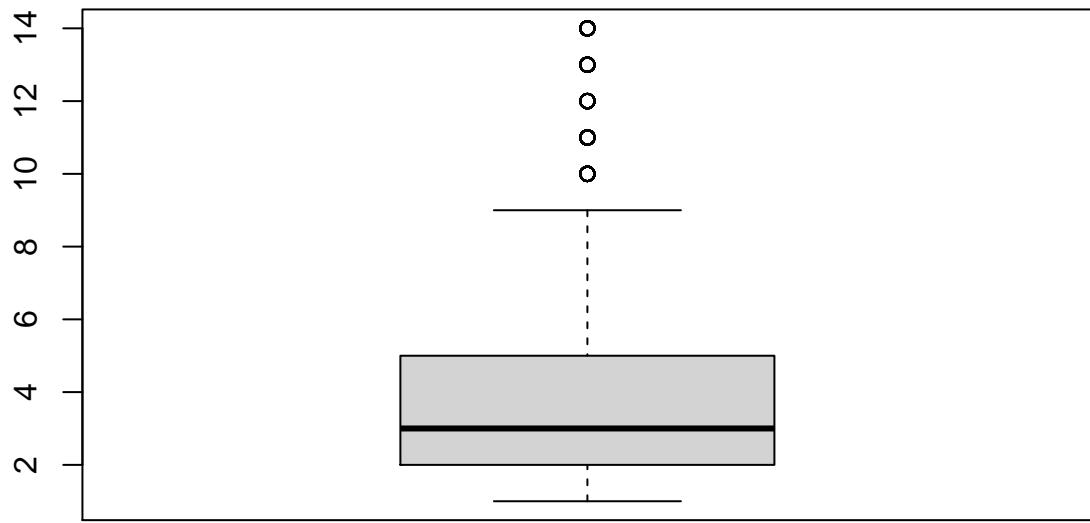


```
#a large amount of orders with very small Sales!
```

```
summary(data$Quantity)
```

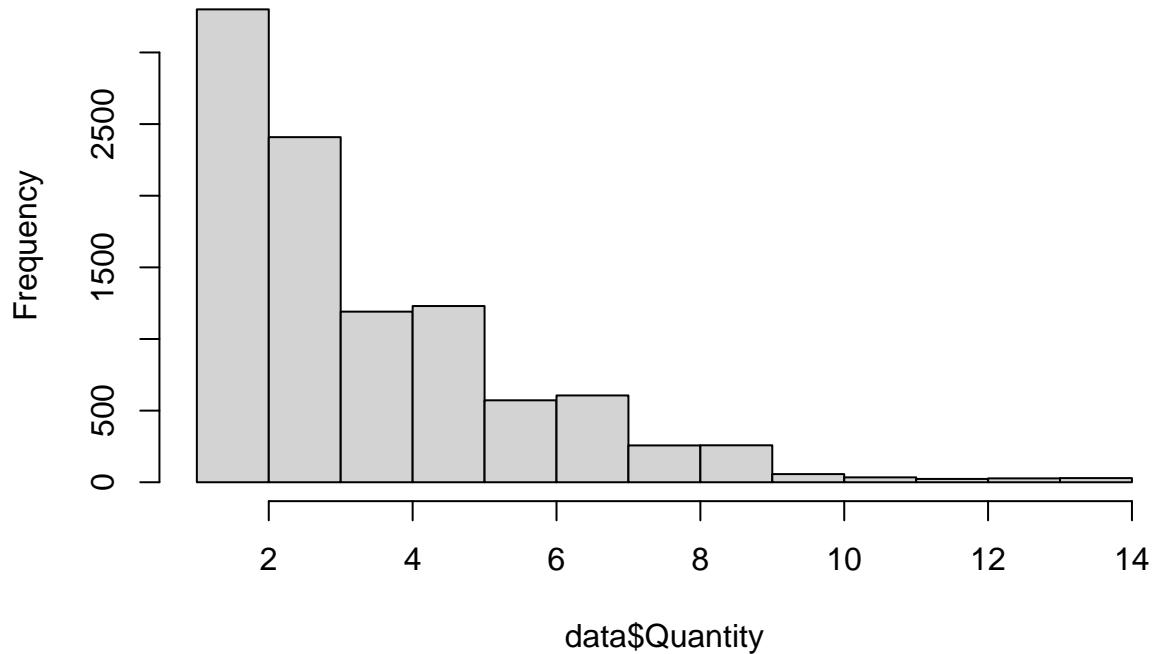
```
##      Min. 1st Qu. Median      Mean 3rd Qu.    Max.
##     1.00    2.00    3.00    3.79    5.00   14.00
```

```
boxplot(data$Quantity)
```



```
#not many outliers - the #of products in each order is stable?  
hist(data$Quantity)
```

Histogram of data\$Quantity

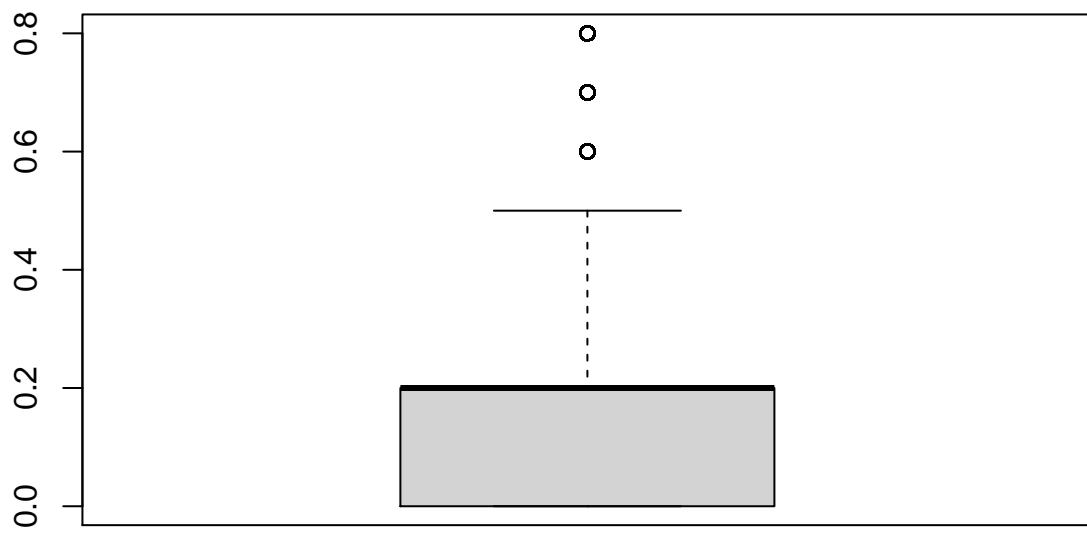


```
#very skewed distribution - most of the orders have small #of items
```

```
summary(data$Discount)
```

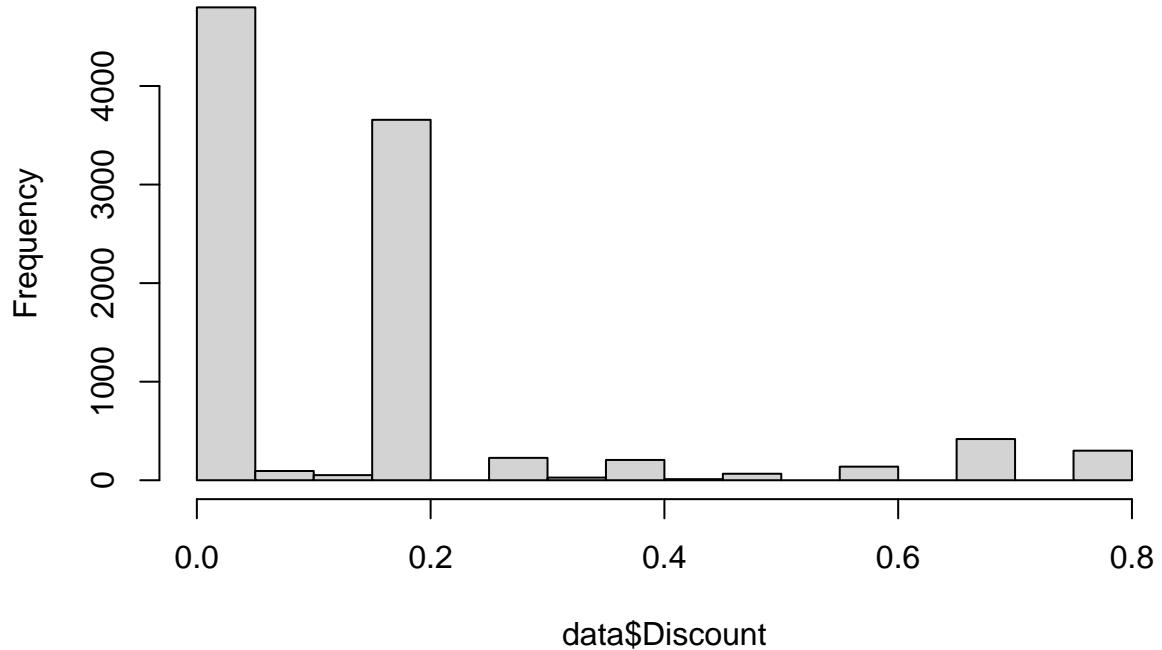
```
##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
## 0.0000 0.0000 0.2000 0.1562 0.2000 0.8000
```

```
boxplot(data$Discount)
```



```
#a strange looking box dataplot? - median & 3rd quantile are the same (0.2) - not many orders have high  
hist(data$Discount)
```

Histogram of data\$Discount

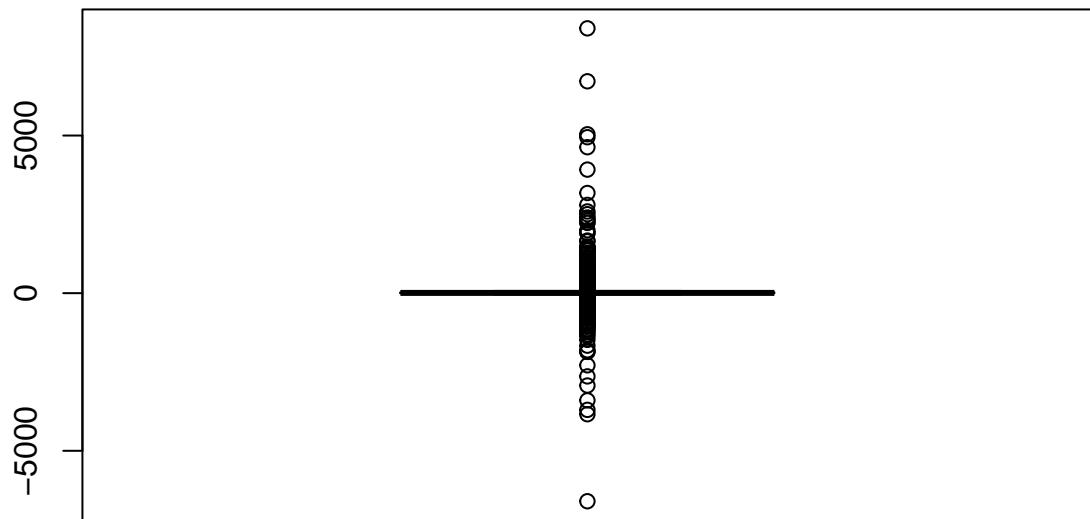


```
#most of the orders were placed without any discounts or with 20% off
```

```
summary(data$Profit)
```

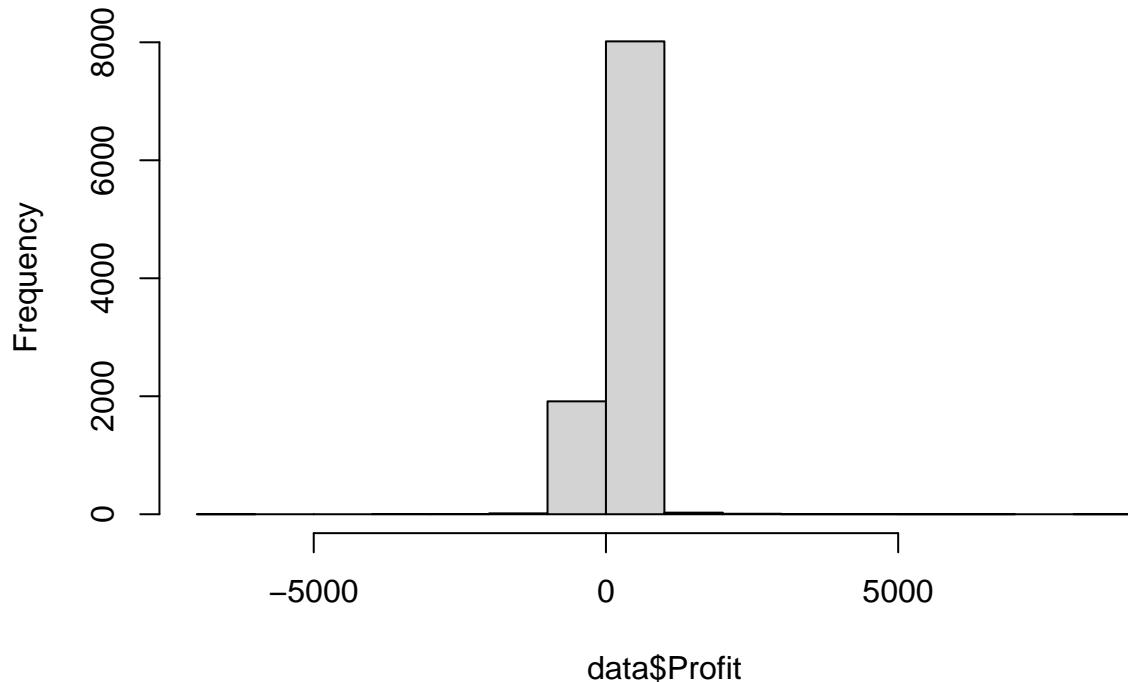
```
##      Min.    1st Qu.     Median      Mean    3rd Qu.      Max.
## -6599.978     1.729     8.666    28.657   29.364  8399.976
```

```
boxplot(data$Profit)
```



```
#most of the profits are outside of the box - but most of them clustered close to the box(not with so e  
hist(data$Profit)
```

Histogram of data\$Profit



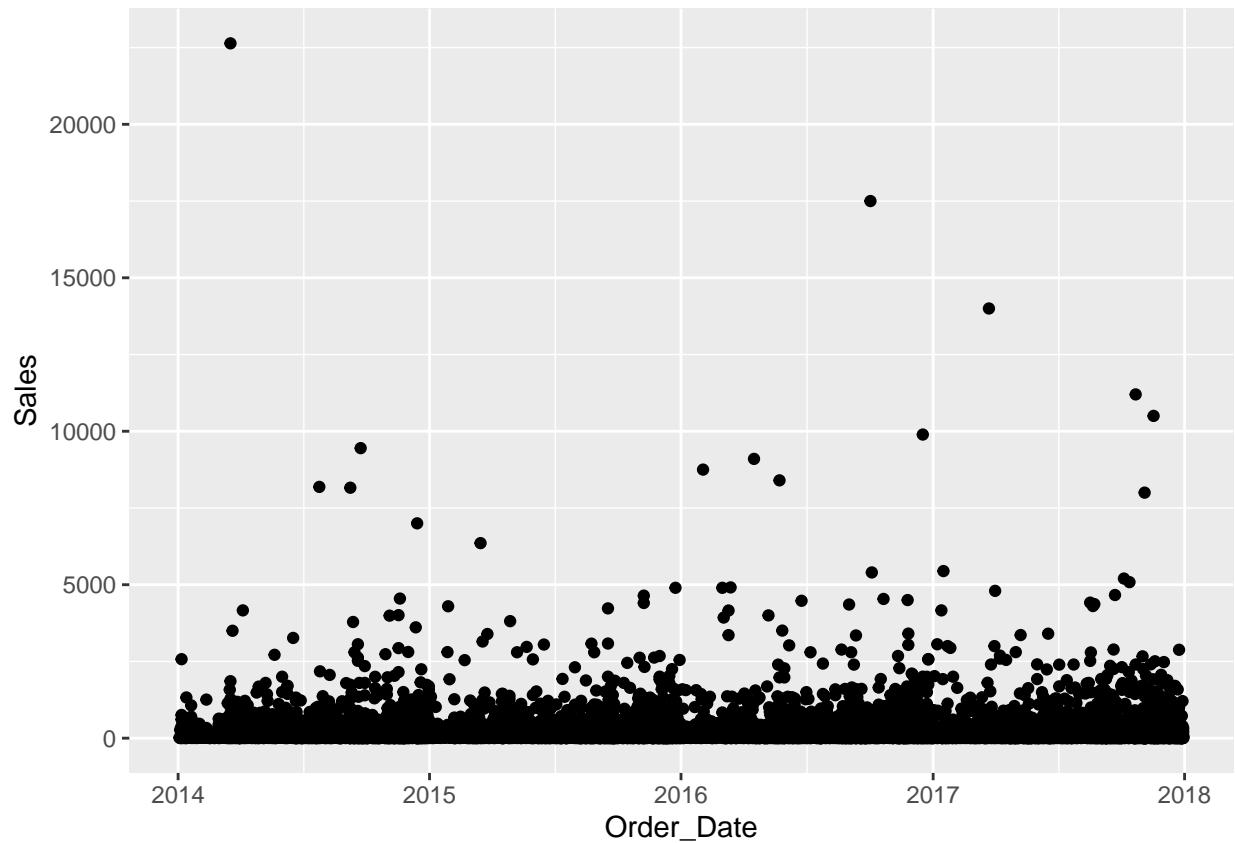
```
#most of the orders have profits ~1000 (or ~800?), and ~ -800
```

Remove the dot in the column names and replace with "_" to make variable names easier to handle:

```
## [1] "i__Row_ID"      "Order_ID"       "Order_Date"     "Ship_Date"  
## [5] "Ship_Mode"       "Customer_ID"    "Customer_Name" "Segment"  
## [9] "Country"         "City"           "State"          "Postal_Code"  
## [13] "Region"          "Product_ID"     "Category"      "Sub_Category"  
## [17] "Product_Name"   "Sales"          "Quantity"      "Discount"  
## [21] "Profit"          "diff_in_days"
```

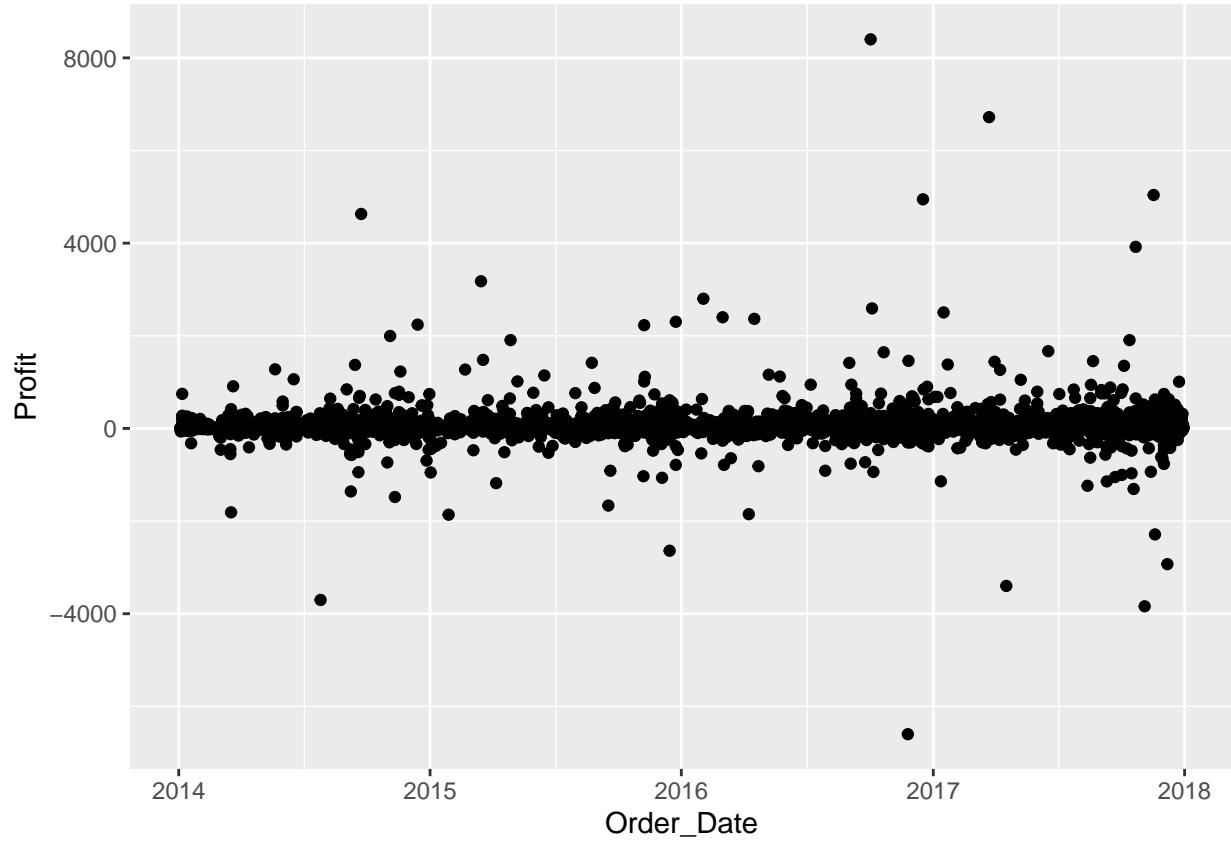
Exploratory Data Analysis

Plot Sales in relation to Order Date:



Plot Profit in relation to Order Date:

```
ggplot(data = data) +  
  geom_point(mapping = aes(x = Order_Date, y = Profit), xlab="Order Date", ylab="Profit")  
  
## Warning: Ignoring unknown parameters: xlab, ylab
```



Some outliers for certain days

```
table(data$`Sub_Category`)
```

```
##
## Accessories Appliances Art Binders Bookcases Chairs
##      775        466    796     1523      228      617
## Copiers Envelopes Fasteners Furnishings Labels Machines
##      68        254     217      957      364      115
## Paper Phones Storage Supplies Tables
##     1370       889     846      190      319
```

look at the time range for these transactions, ie. start date for Order_Date column:

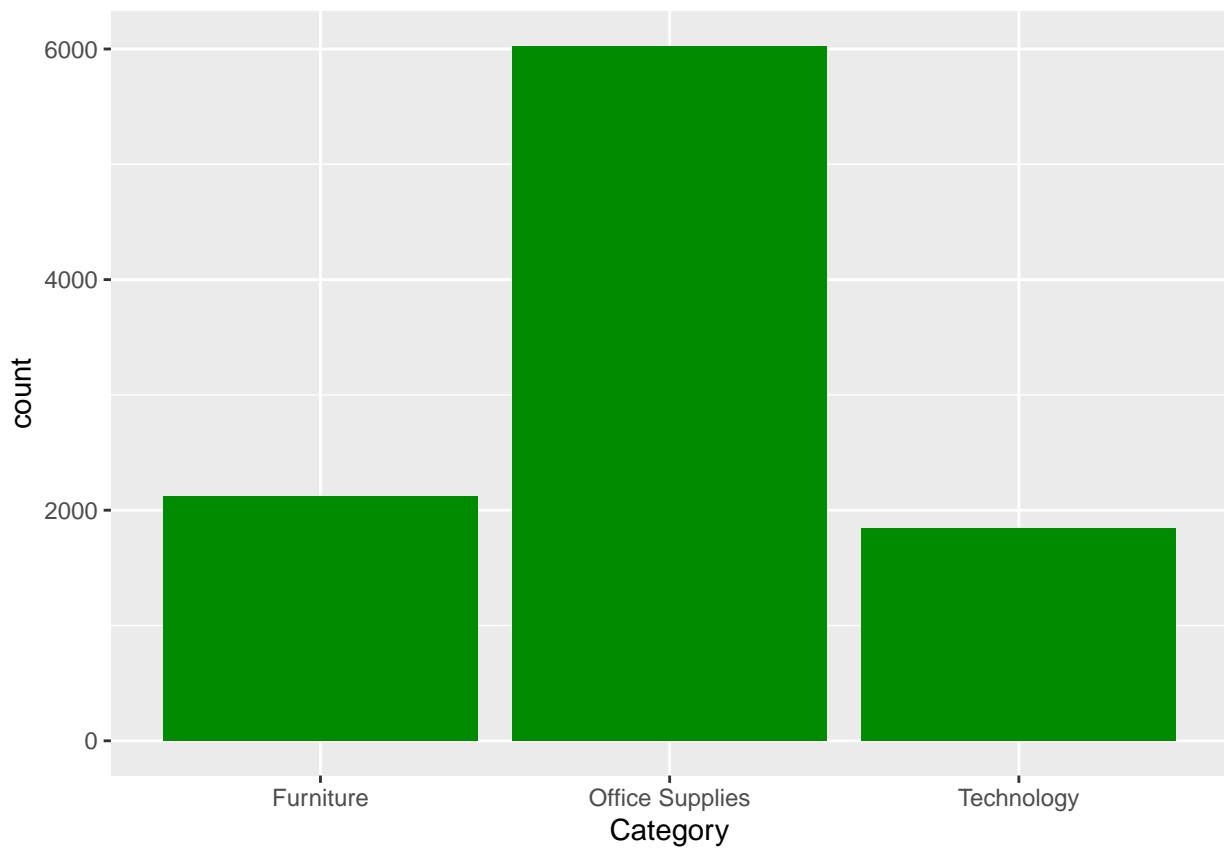
```
summary(data$Order_Date)
```

```
##           Min.     1st Qu.    Median      Mean     3rd Qu.      Max.
## "2014-01-03" "2015-05-23" "2016-06-26" "2016-04-30" "2017-05-14" "2017-12-30"
```

```
# [1] min "2014-01-03", max "2017-12-30"
```

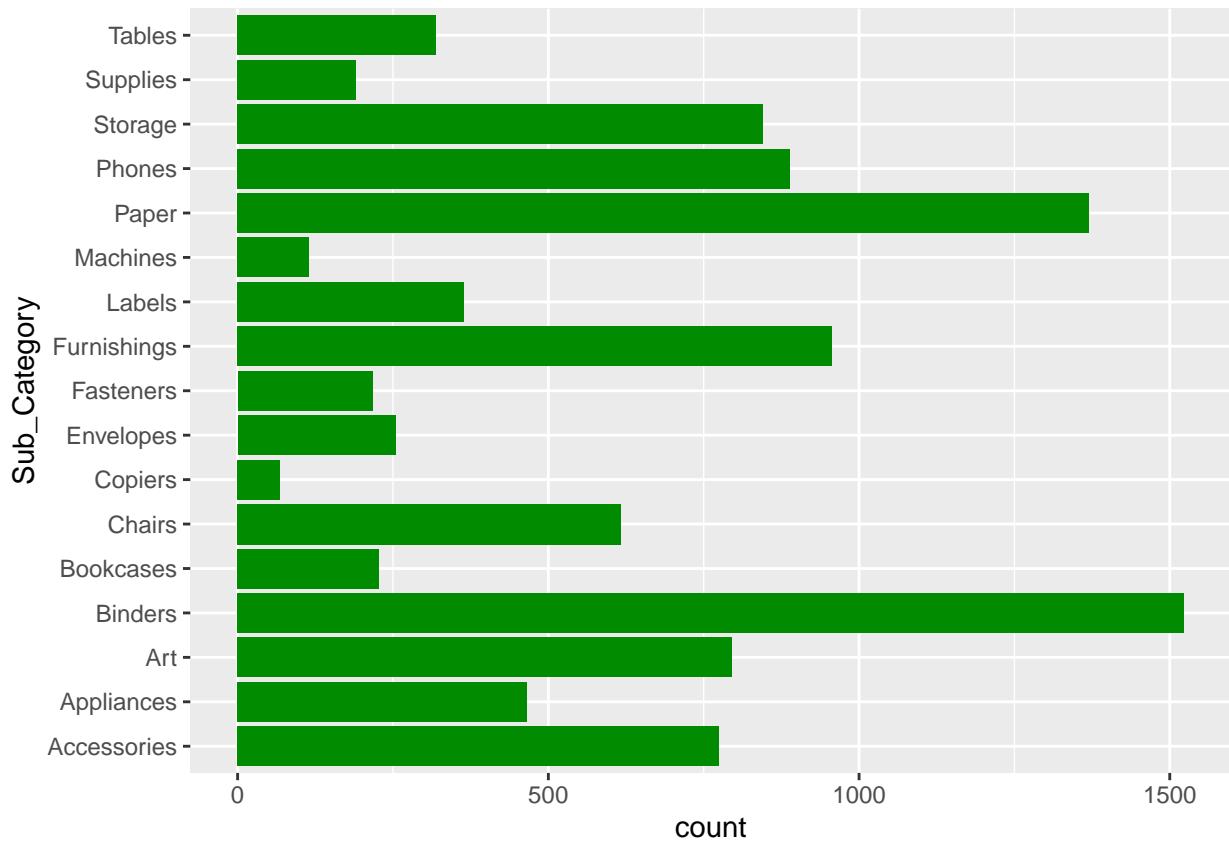
Basically this dataset covers transactions ranging from 2014-01-03 to 2017-12-30.

```
ggplot(data = data) +  
  geom_bar(mapping = aes(x = Category), fill="green4")
```



Most type of products sold belong to the Office supplies category.

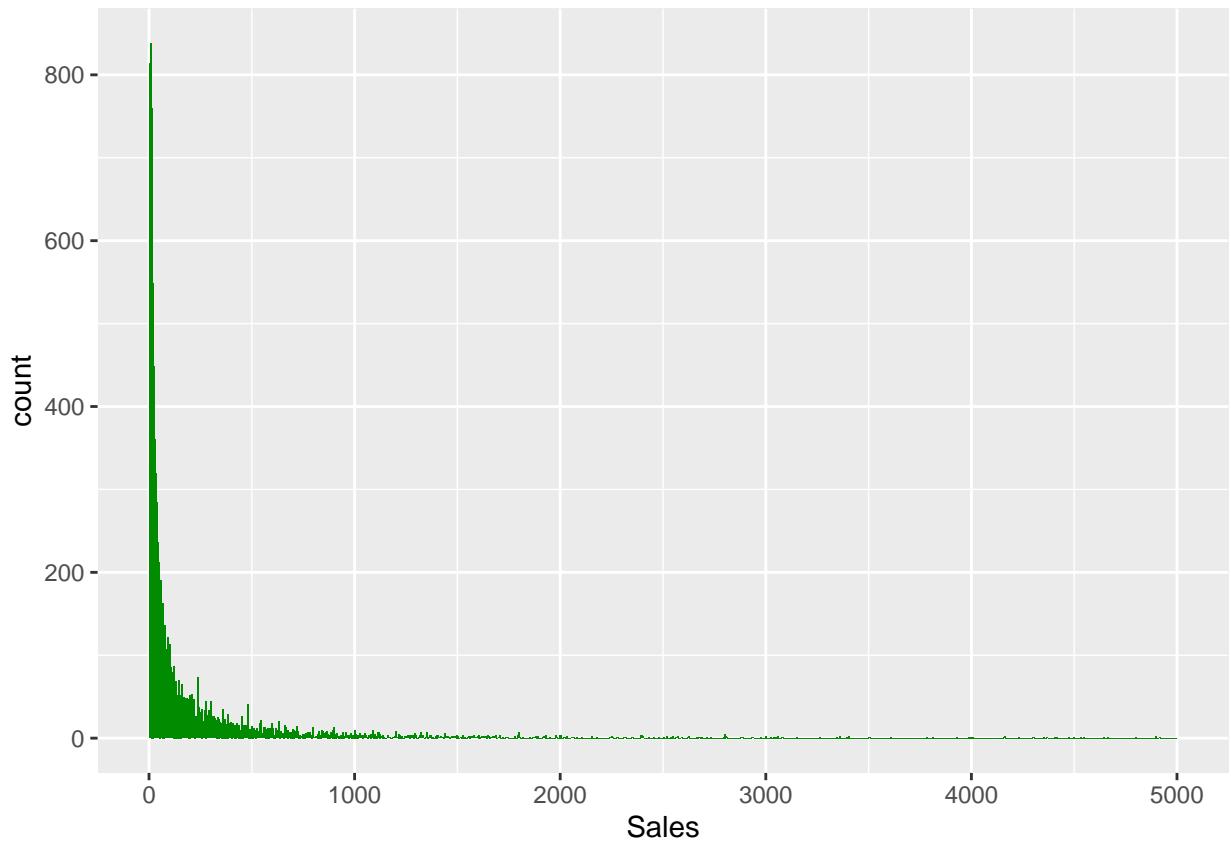
```
ggplot(data = data) +  
  geom_bar(mapping = aes(y = `Sub_Category`), fill="green4")
```



```
ggplot(data = data, mapping = aes(x = Sales)) +  
  xlim(0, 5000) +  
  geom_histogram(binwidth = 5, fill="green4")
```

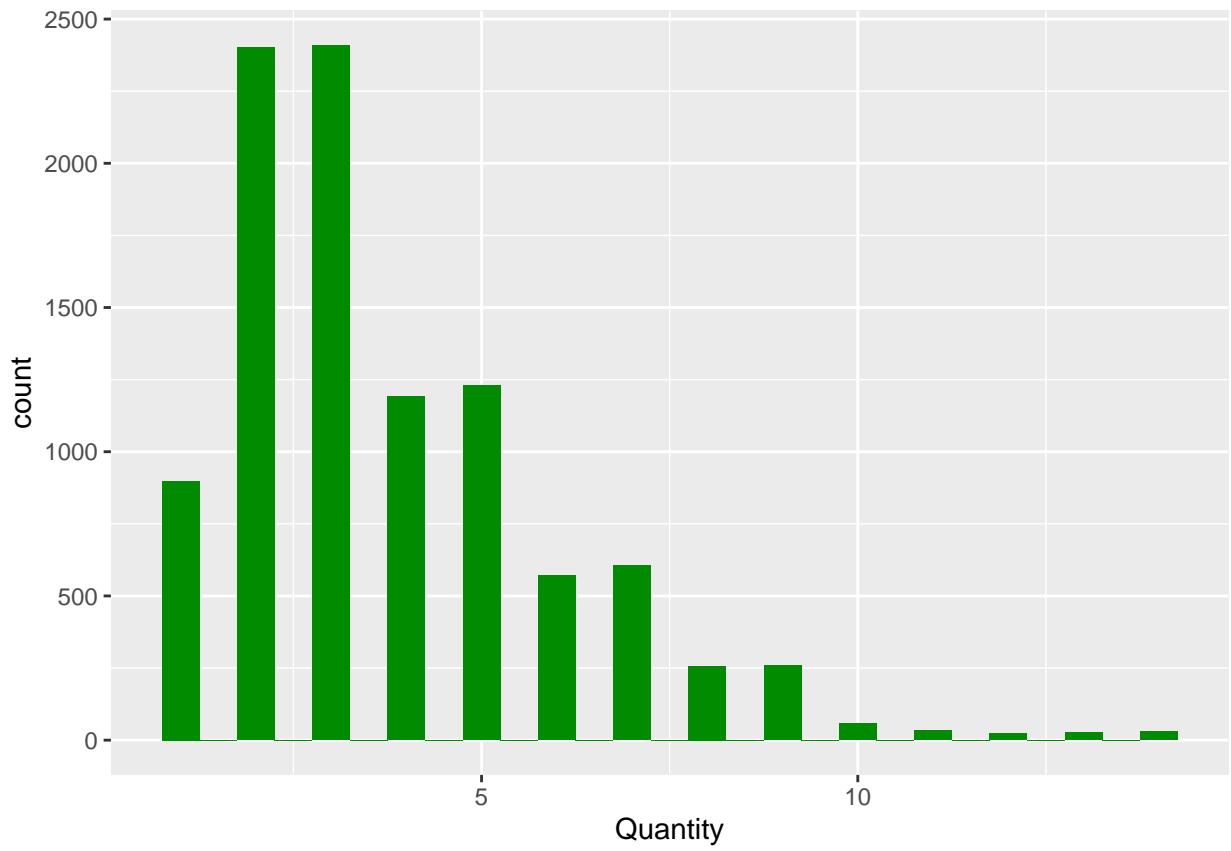
```
## Warning: Removed 19 rows containing non-finite values (stat_bin).
```

```
## Warning: Removed 2 rows containing missing values (geom_bar).
```

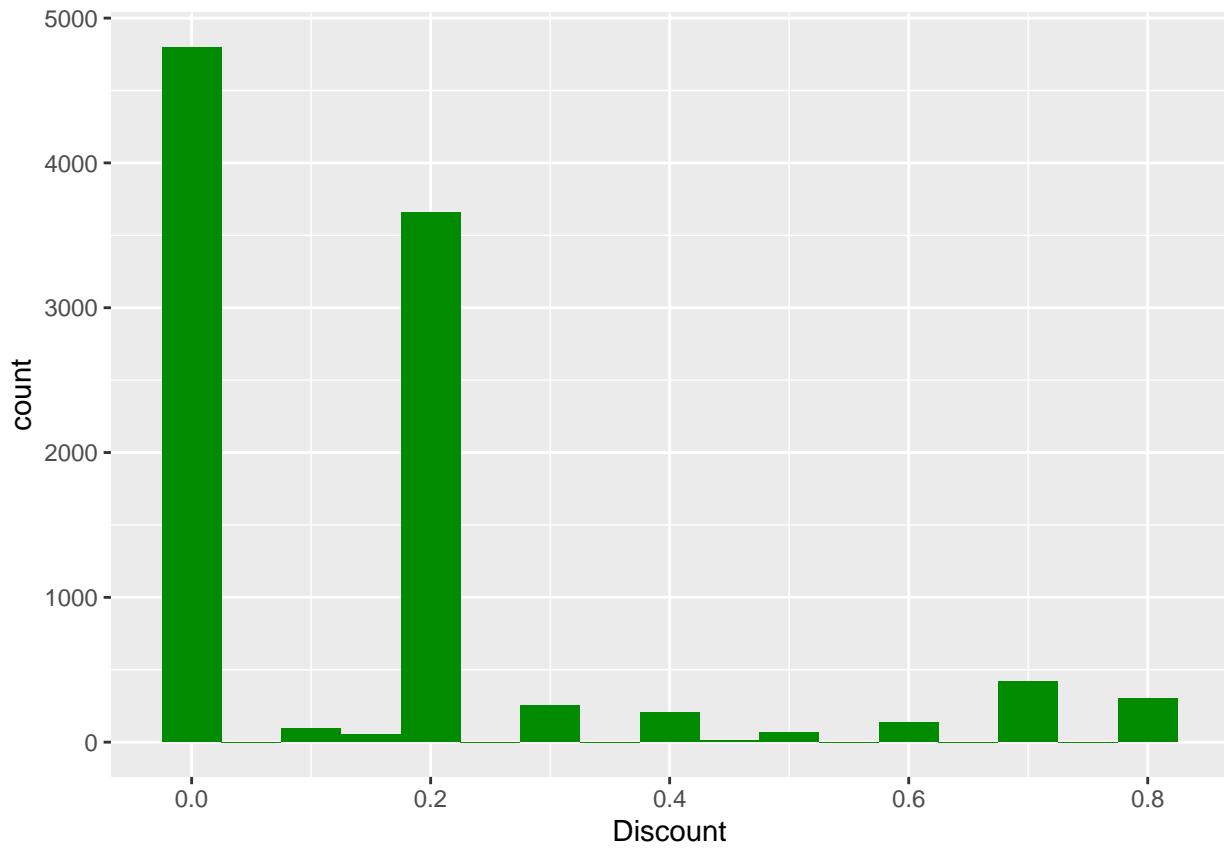


Most sales are very few items (<500).

```
ggplot(data = data, mapping = aes(x = Quantity)) +  
  geom_histogram(binwidth = 0.5, fill="green4")
```



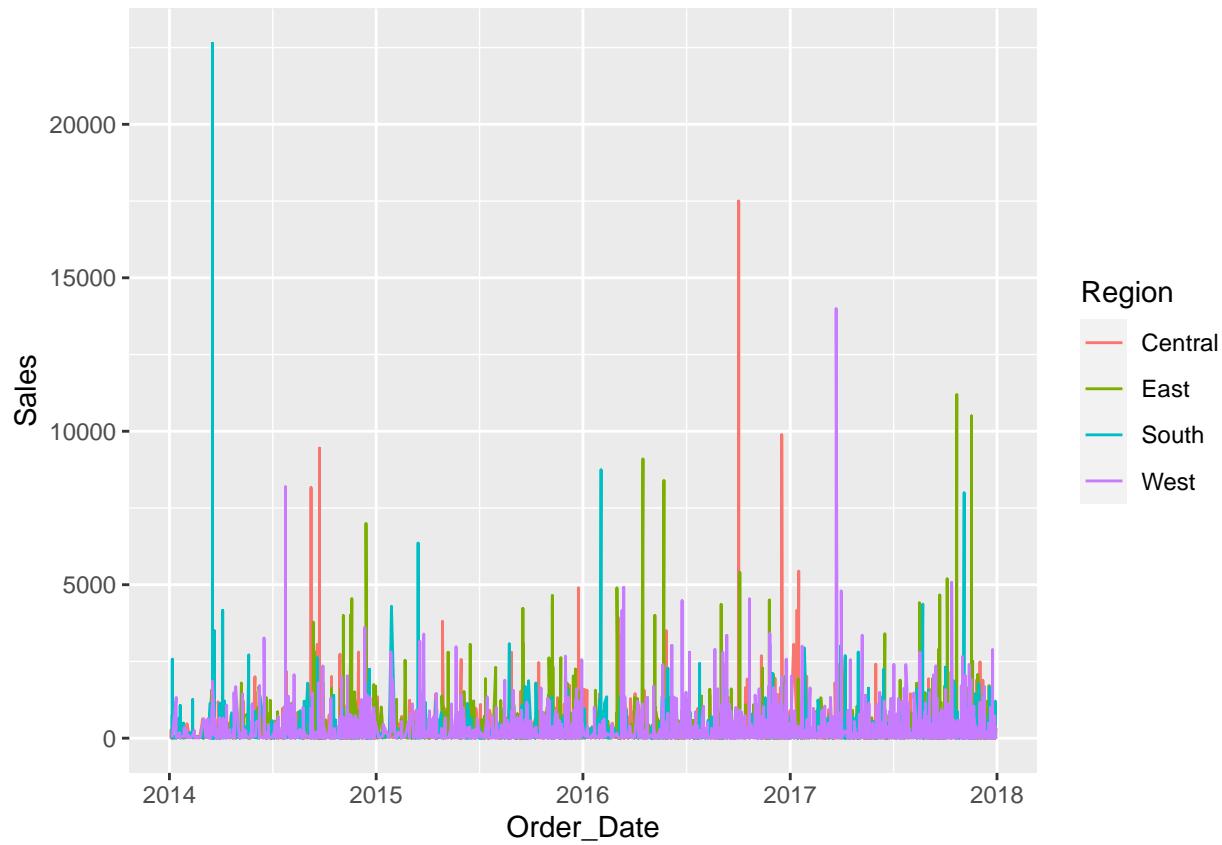
```
ggplot(data = data) +  
  geom_histogram(mapping = aes(x = Discount),  
                 binwidth = 0.05,  
                 xlab="Discount",  
                 fill="green4")  
  
## Warning: Ignoring unknown parameters: xlab
```



Sales transactions mostly do not involve discounts.

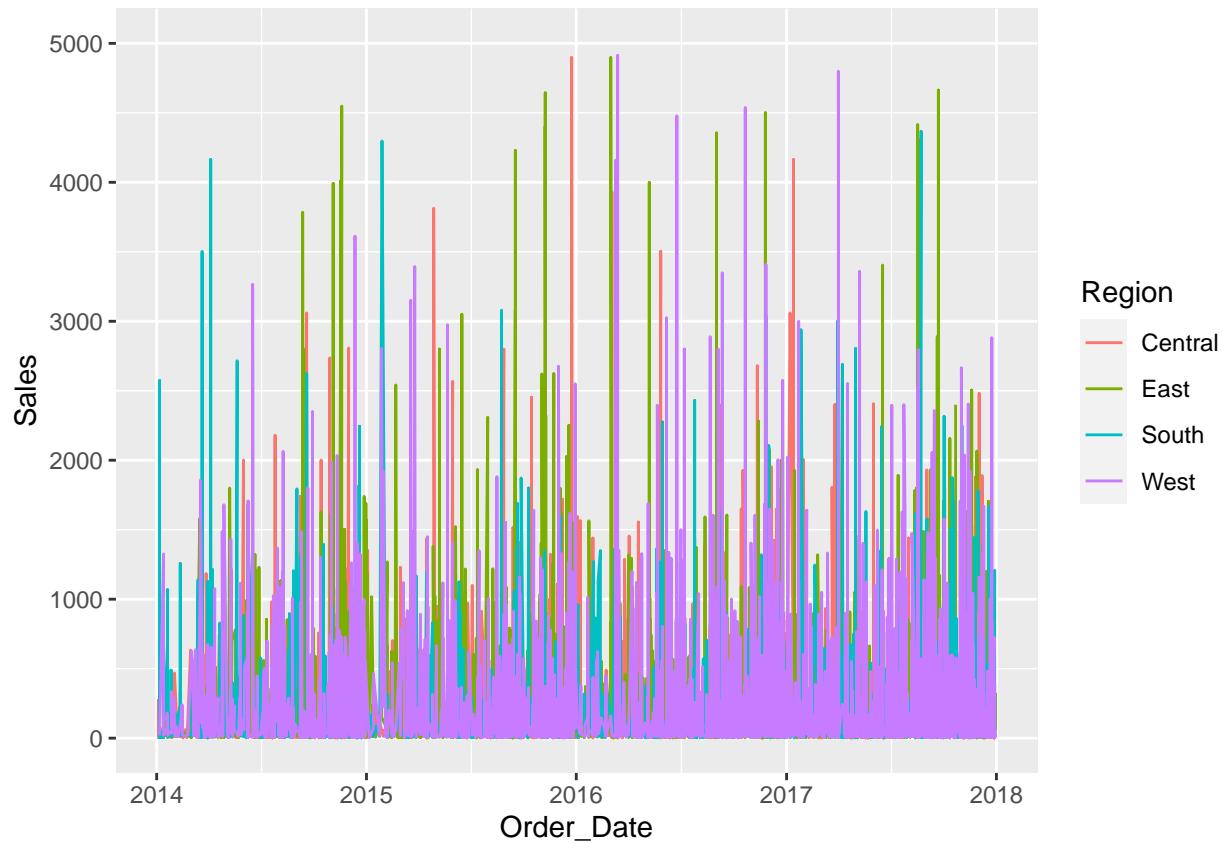
Visualise sales transactions by Region over time (order date).

```
ggplot(data, aes(Order_Date, Sales, color=Region)) +  
  geom_line()
```



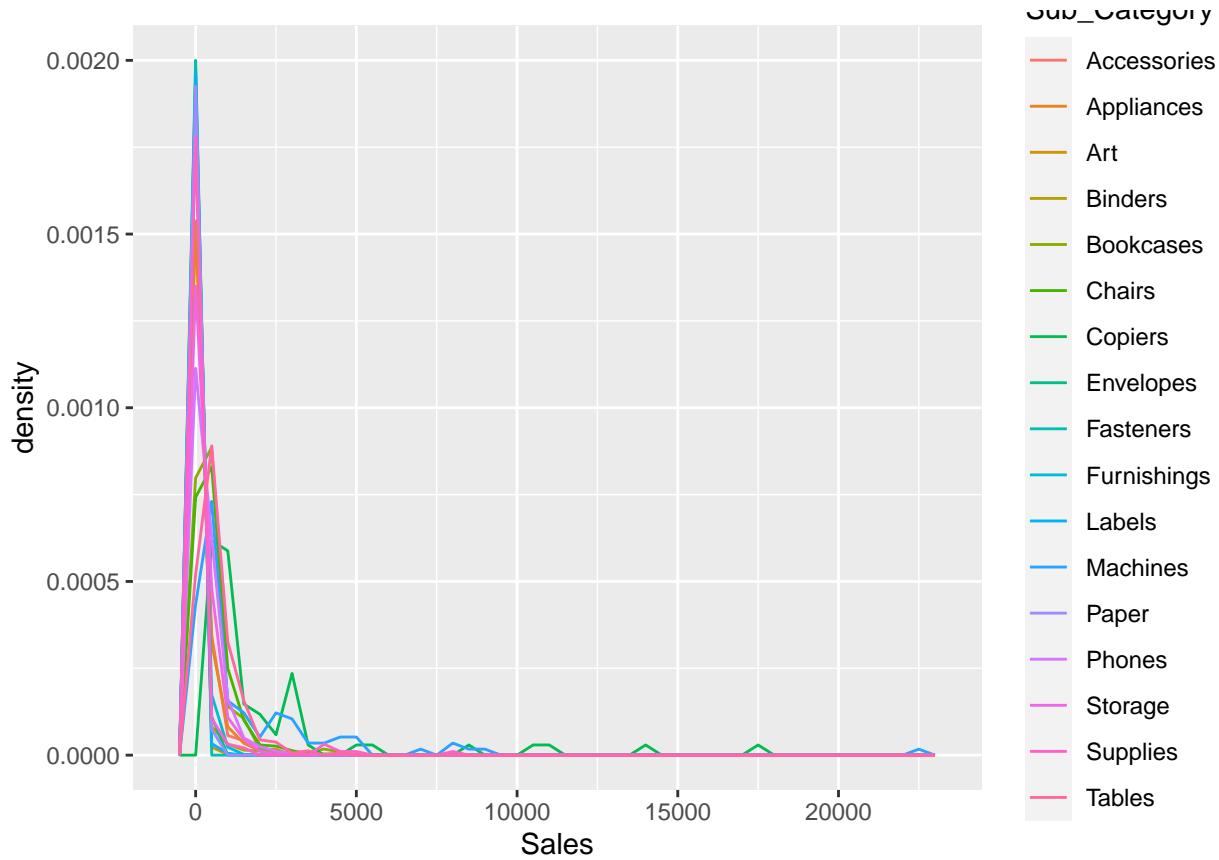
Let's zoom in a little bit - Visualise sales transactions by Region over time (order date).

```
ggplot(data, aes(Order_Date, Sales,color=Region)) +
  geom_line() +
  ylim(0,5000)
```



How does profit change with sub-category?

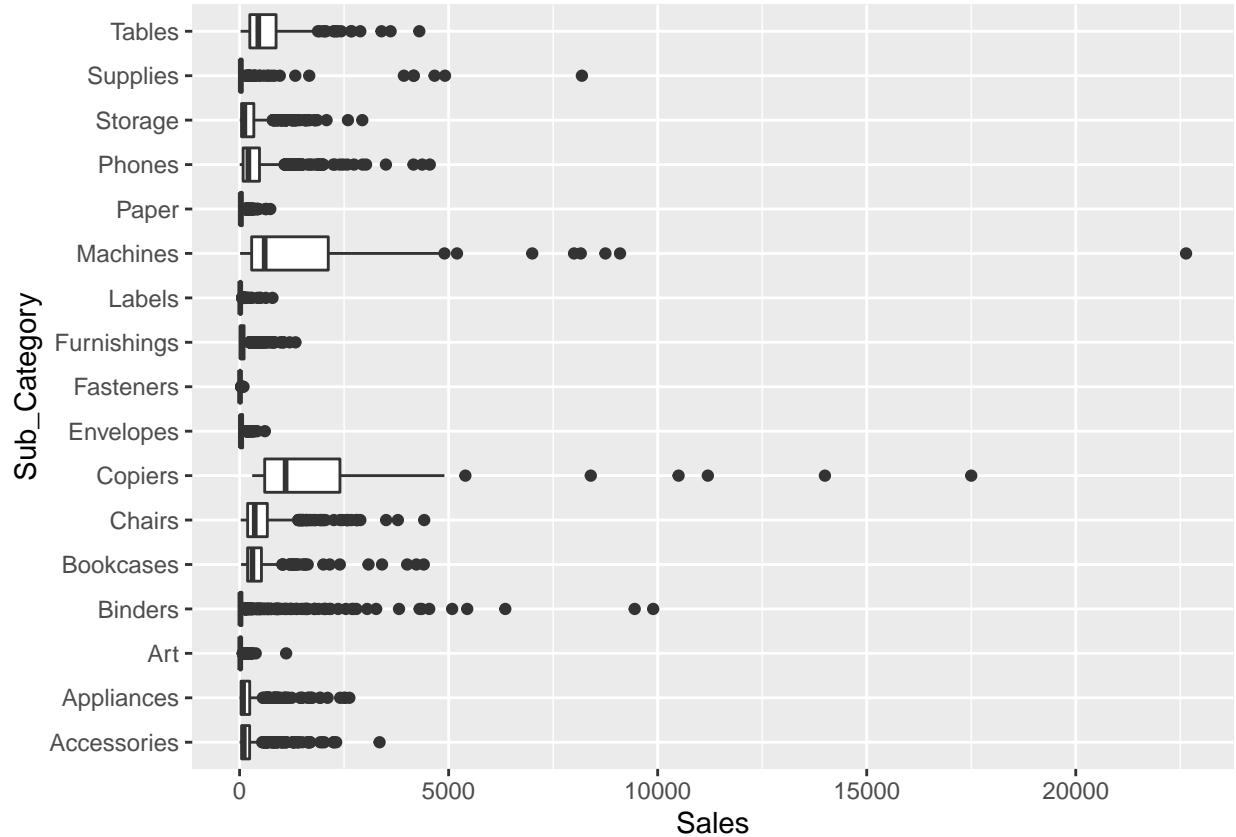
```
#density plot where the count is standardized,area under each frequency is 1
ggplot(data = data, mapping = aes(x = Sales, y = ..density..)) +
  geom_freqpoly(mapping = aes(colour = Sub_Category), binwidth = 500)
```



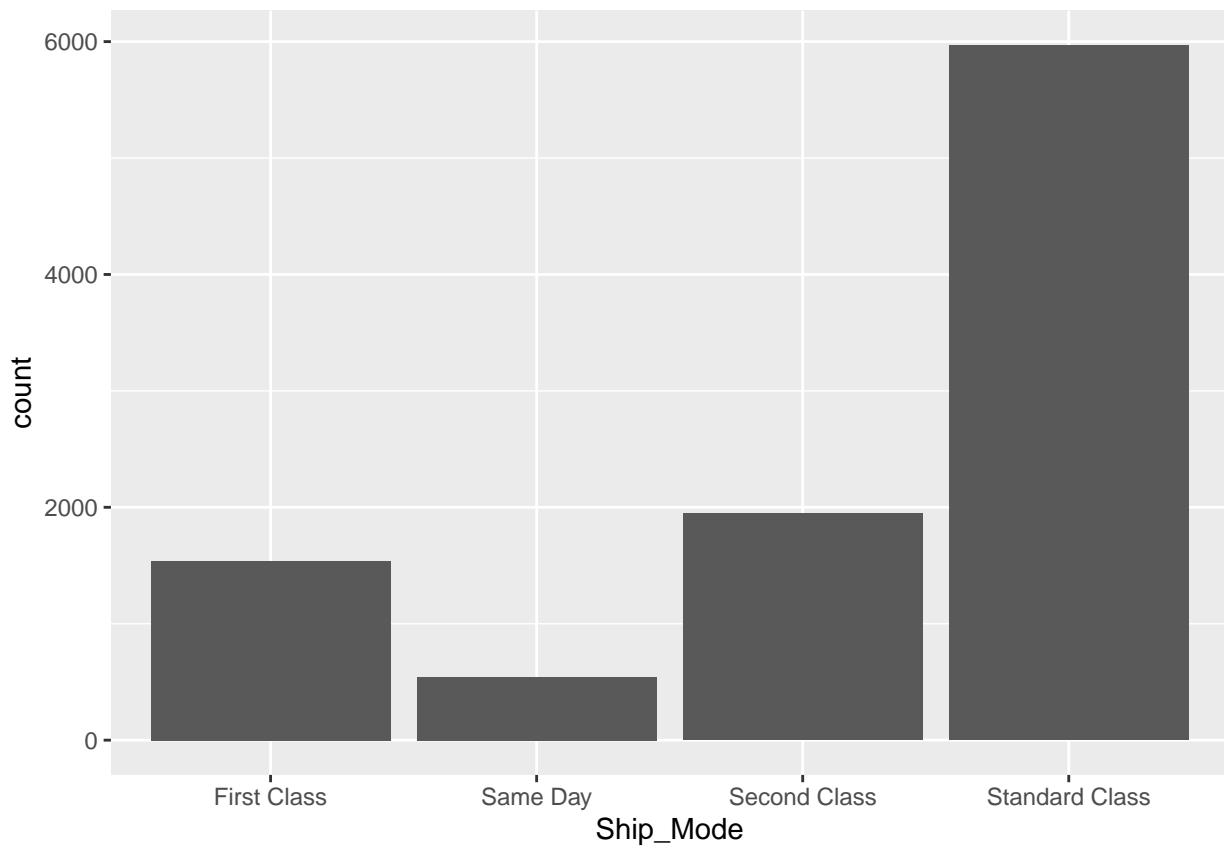
It looks like some categories of items ie. supplies or accessories have negative sales values.

How does sales vary across sub category?

```
ggplot(data = data, mapping = aes(x = Sales, y = `Sub_Category`)) +
  geom_boxplot()
```



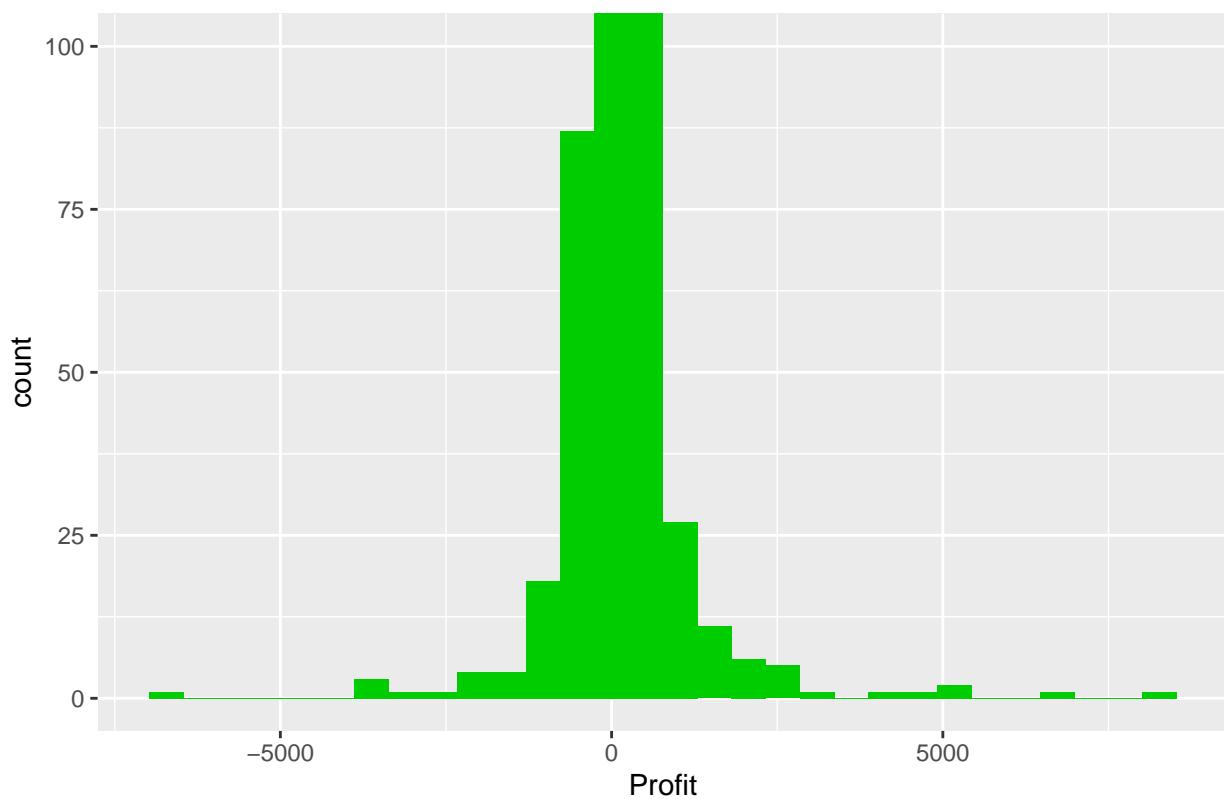
```
ggplot(data = data, mapping = aes(x = Ship_Mode)) +  
  geom_bar()
```



Most transactions are shipped via Standard Class method.

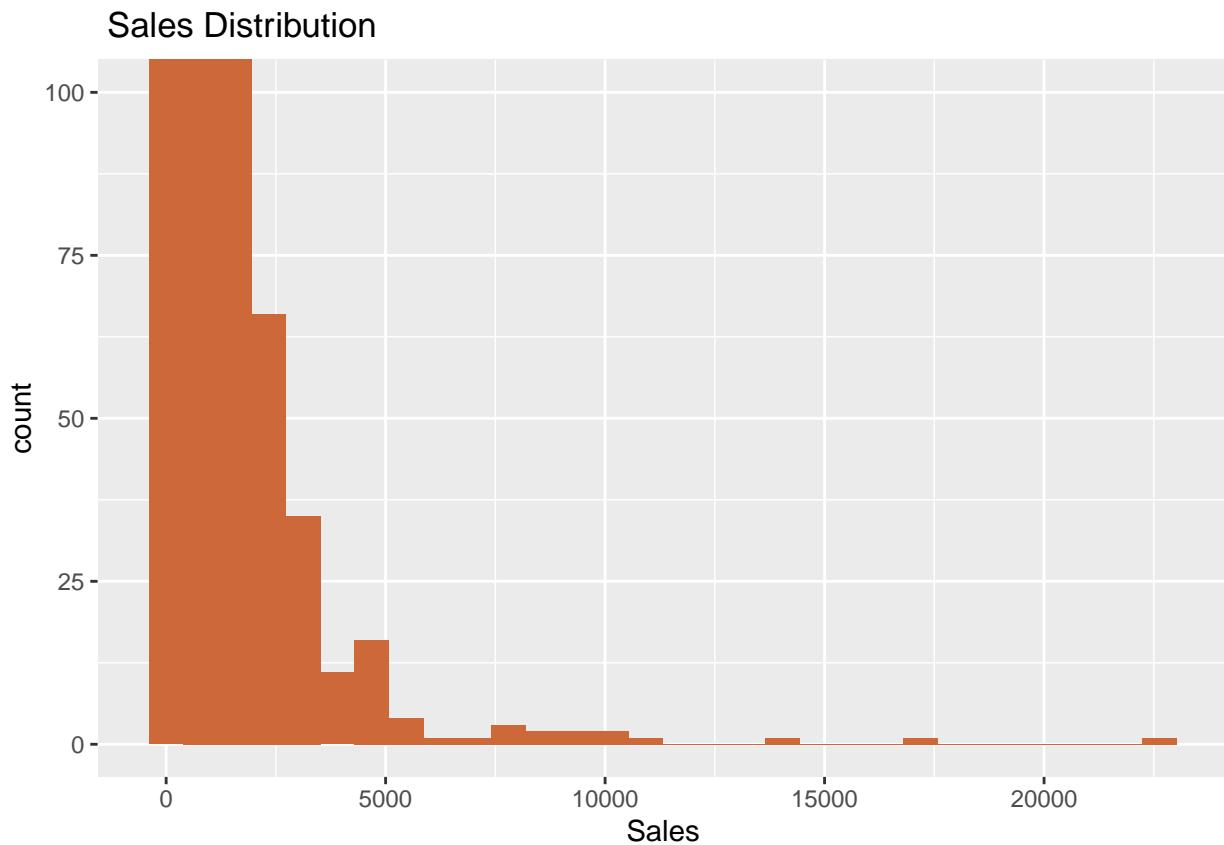
```
ggplot(data)+  
  geom_histogram(mapping=aes(x=Profit), fill="green3") +  
  coord_cartesian(ylim = c(0, 100)) +  
  labs(title=" Profit Distribution")  
  
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```

Profit Distribution



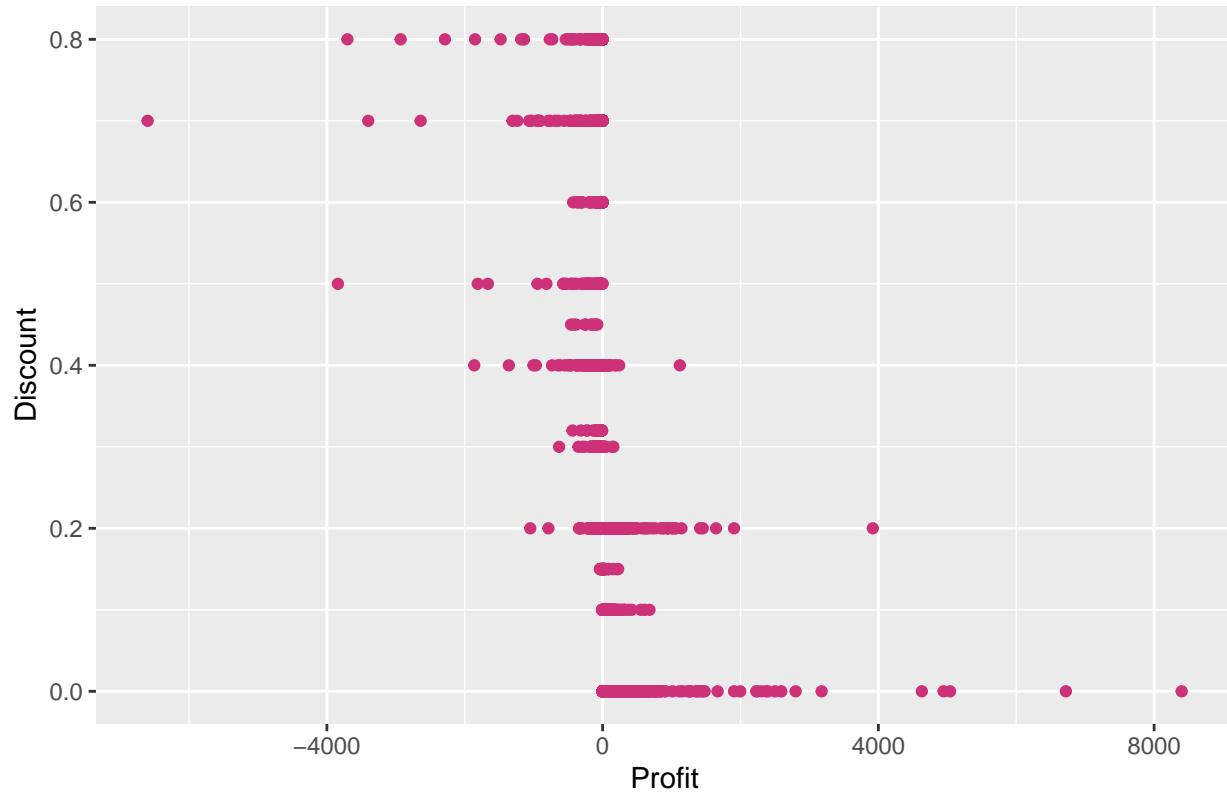
```
ggplot(data)+  
  geom_histogram(mapping=aes(x=Sales),fill="sienna3") +  
  coord_cartesian(ylim = c(0, 100)) + labs(title=" Sales Distribution")
```

```
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```



```
ggplot(data) +  
  geom_point(mapping = aes(x = Profit, y = Discount), colour="violetred3") +  
  labs(title=" Profit Discount Distribution")
```

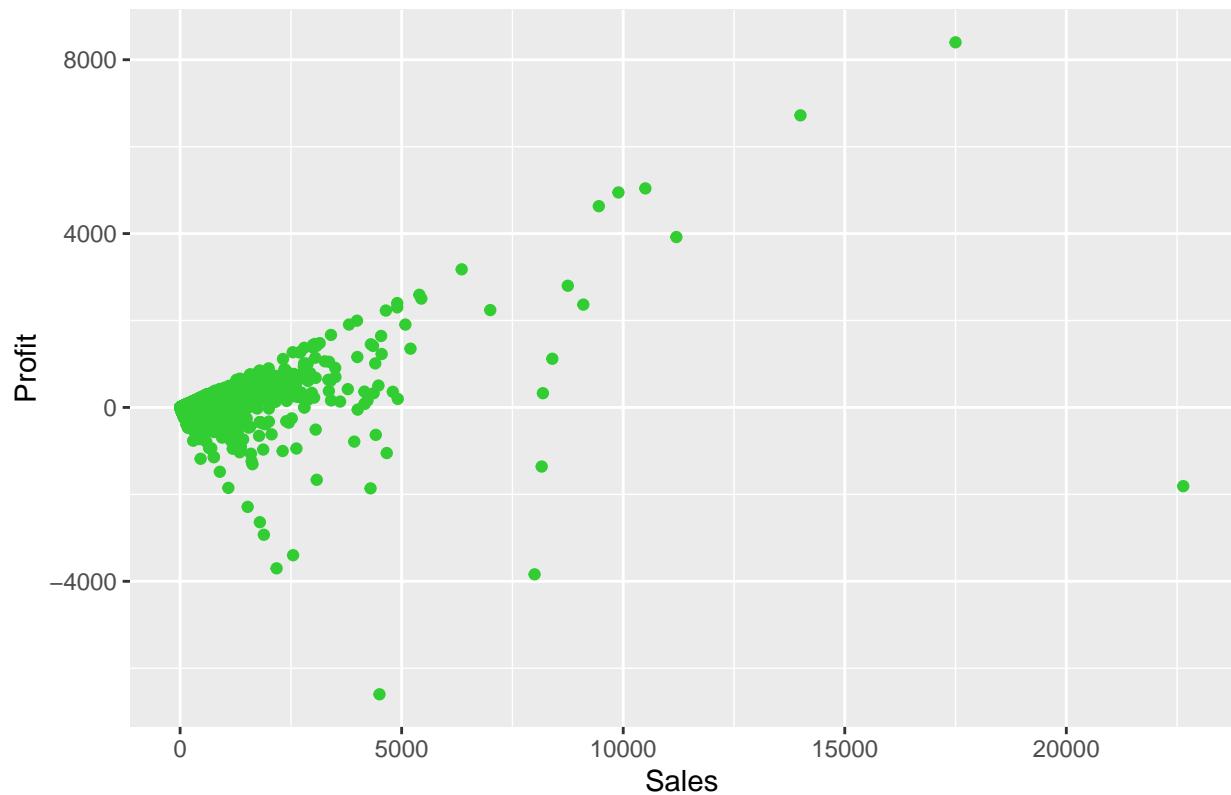
Profit Discount Distribution



Sales Profit

```
ggplot(data) +  
  geom_point(mapping = aes(x = Sales, y = Profit), colour="limegreen") +  
  labs(title=" Sales Profit Distribution")
```

Sales Profit Distribution



```
#product name and product id mismatch
data %>%
  distinct(Product_Name, Product_ID) %>%
  group_by(Product_ID) %>%
  filter(n() > 1) %>%
  select(Product_ID)
```

```
## # A tibble: 64 x 1
## # Groups:   Product_ID [32]
##   Product_ID
##   <chr>
## 1 FUR-FU-10004848
## 2 FUR-CH-10001146
## 3 OFF-BI-10004654
## 4 FUR-CH-10001146
## 5 OFF-PA-10002377
## 6 OFF-AR-10001149
## 7 OFF-PA-10000659
## 8 TEC-MA-10001148
## 9 FUR-FU-10004017
## 10 TEC-AC-10003832
## # ... with 54 more rows
```

```
#total category and subcategory
```

```

count_category<-unique(data$Category)
length(count_category)

## [1] 3

count_subcategory<-unique(data$Sub_Category)
length(count_subcategory)

## [1] 17

data %>%
  distinct(Category, Sub_Category)

##          Category Sub_Category
## 1      Furniture    Bookcases
## 2      Furniture        Chairs
## 3 Office Supplies       Labels
## 4      Furniture        Tables
## 5 Office Supplies     Storage
## 6      Furniture   Furnishings
## 7 Office Supplies         Art
## 8      Technology      Phones
## 9 Office Supplies      Binders
## 10 Office Supplies   Appliances
## 11 Office Supplies        Paper
## 12      Technology   Accessories
## 13 Office Supplies      Envelopes
## 14 Office Supplies     Fasteners
## 15 Office Supplies      Supplies
## 16      Technology      Machines
## 17      Technology      Copiers

superstore_sales<-data %>%
  select(Order_Date,Sales)

superstore_sales<-as_tibble(superstore_sales)

```

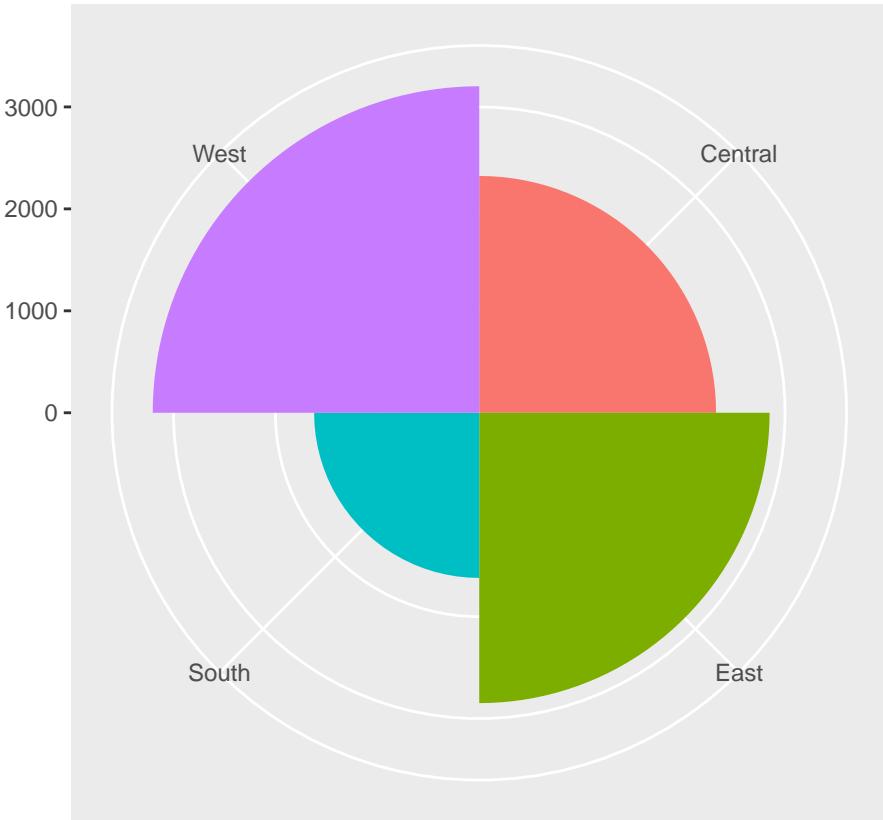
Transactions by region:

```

bar <- ggplot(data = data) +
  geom_bar(
    mapping = aes(x = Region, fill = Region),
    show.legend = FALSE,
    width = 1
  ) +
  theme(aspect.ratio = 1) +
  labs(x = NULL, y = NULL)

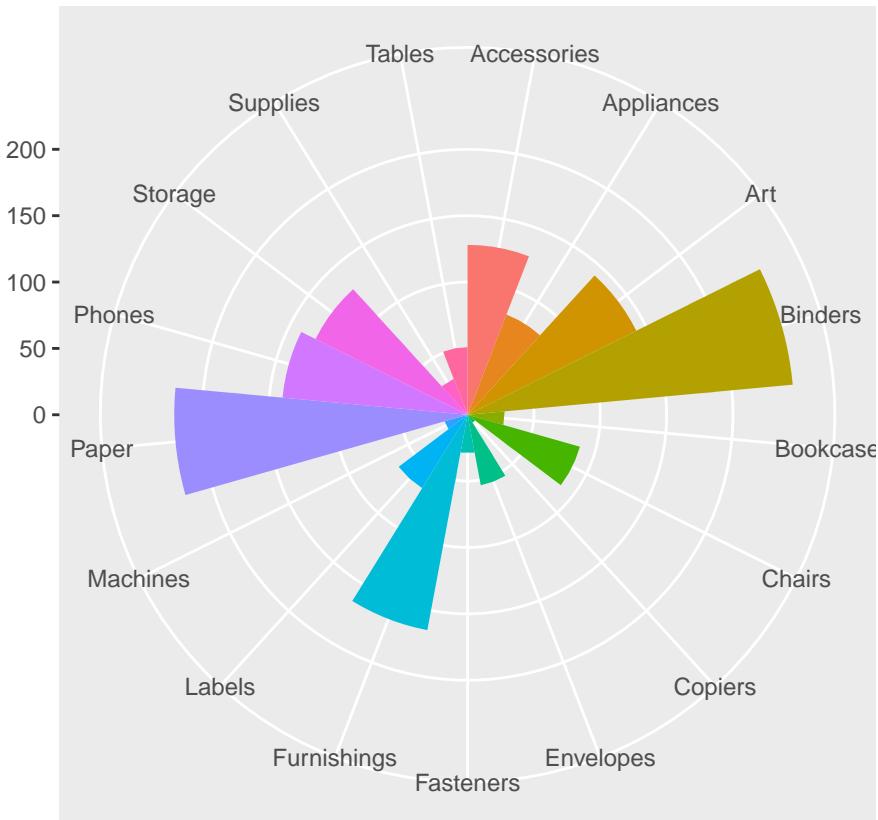
bar + coord_polar()

```



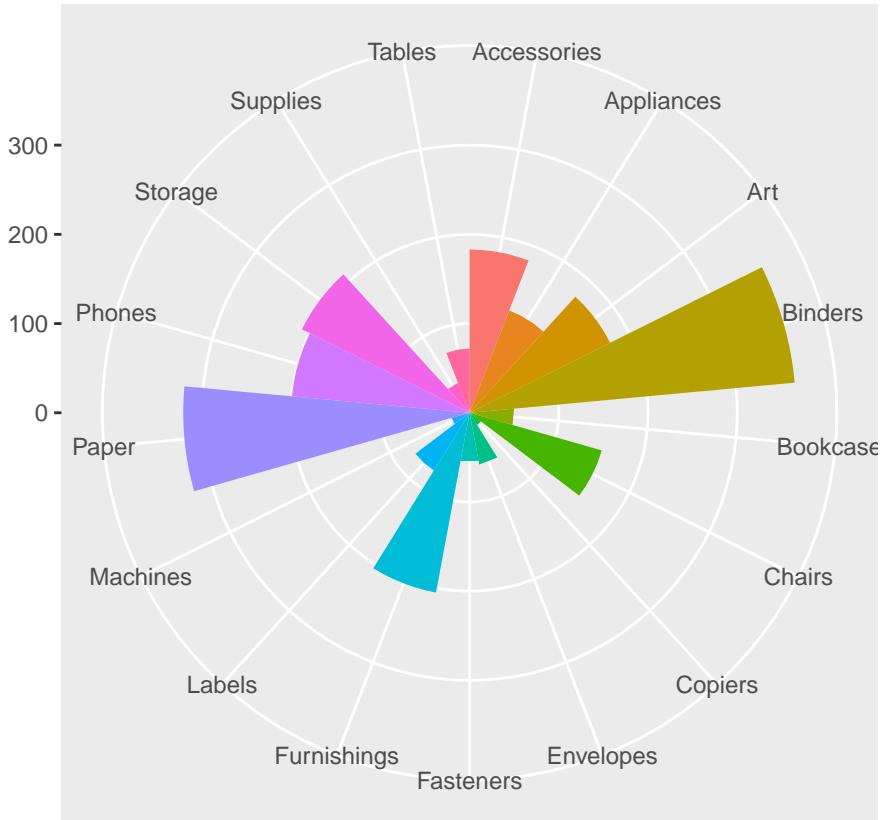
The above chart shows proportions of transactions from the different regions.

```
#Extracting the rows for South region, and sub-categories:  
South <- data %>%  
  select(Region, Sub_Category) %>%  
  filter(Region == "South")  
  
bar <- ggplot(data = South) +  
  geom_bar(  
    mapping = aes(x = Sub_Category, fill = Sub_Category),  
    show.legend = FALSE,  
    width = 1  
  ) +  
  theme(aspect.ratio = 1) +  
  labs(x = NULL, y = NULL)  
  
bar + coord_polar()
```



In the South, most transactions are Binders, Paper, or Furnishings.

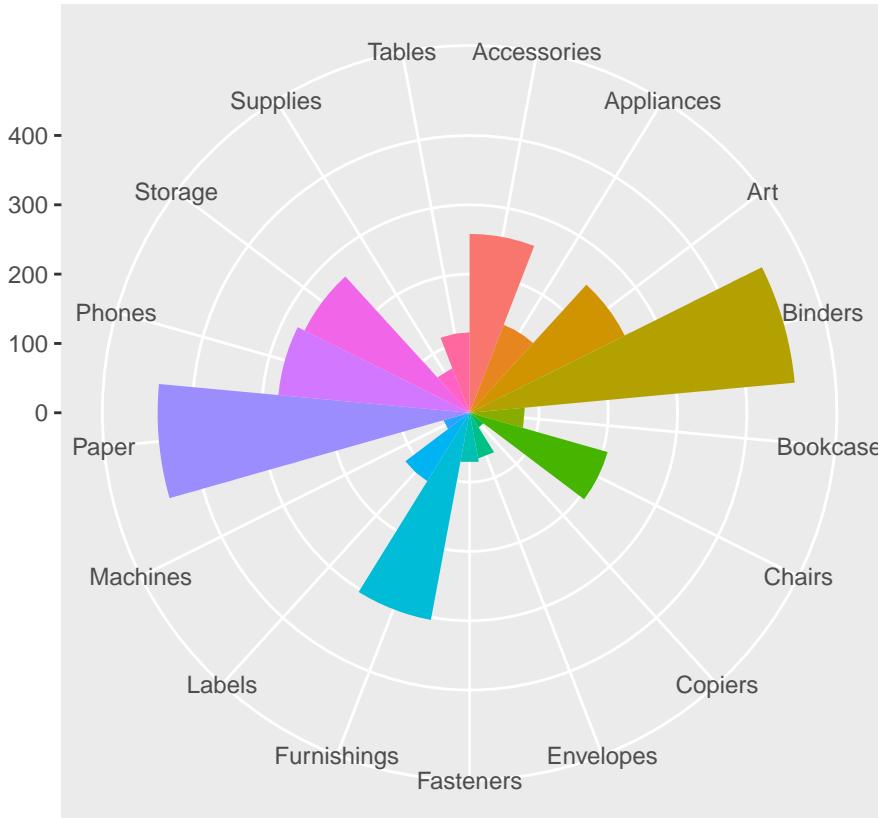
```
#Extracting the rows for Central region, and sub-categories:  
Central <- data %>%  
  select(Region, Sub_Category) %>%  
  filter(Region == "Central")  
  
bar <- ggplot(data = Central) +  
  geom_bar(  
    mapping = aes(x = Sub_Category, fill = Sub_Category),  
    show.legend = FALSE,  
    width = 1  
) +  
  theme(aspect.ratio = 1) +  
  labs(x = NULL, y = NULL)  
  
bar + coord_polar()
```



```
#Extracting the rows for West region, and sub-categories:
West <- data %>%
  select(Region, Sub_Category) %>%
  filter(Region == "West")

bar <- ggplot(data = West) +
  geom_bar(
    mapping = aes(x = Sub_Category, fill = Sub_Category),
    show.legend = FALSE,
    width = 1
  ) +
  theme(aspect.ratio = 1) +
  labs(x = NULL, y = NULL)

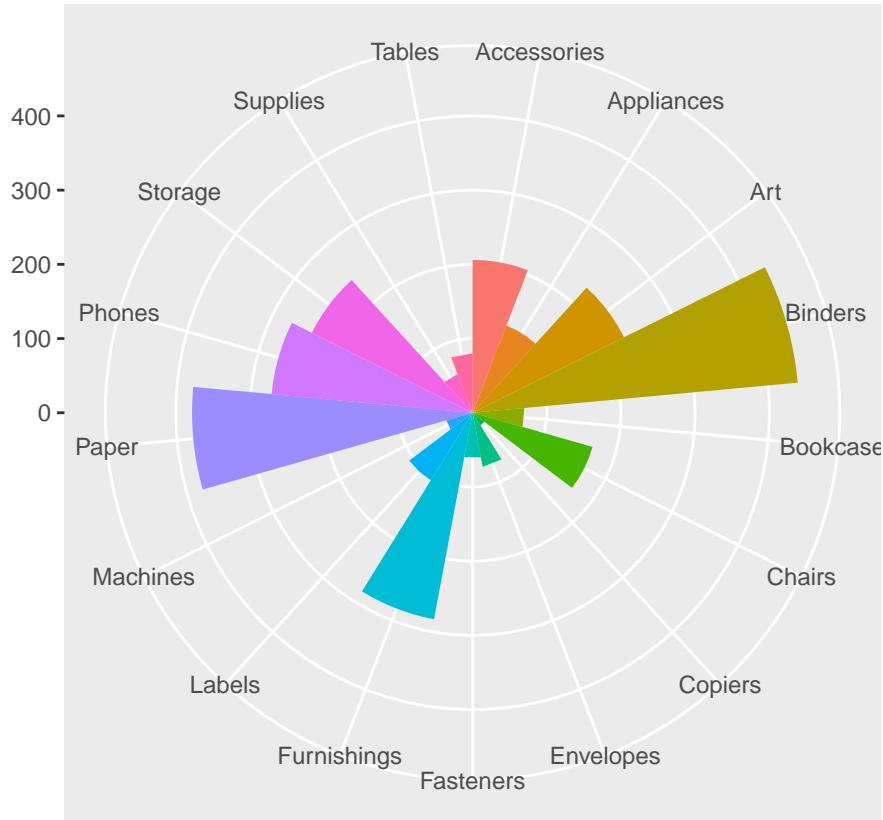
bar + coord_polar()
```



```
#Extracting the rows for East region, and sub-categories:
East <- data %>%
  select(Region, Sub_Category) %>%
  filter(Region == "East")

bar <- ggplot(data = East) +
  geom_bar(
    mapping = aes(x = Sub_Category, fill = Sub_Category),
    show.legend = FALSE,
    width = 1
  ) +
  theme(aspect.ratio = 1) +
  labs(x = NULL, y = NULL)

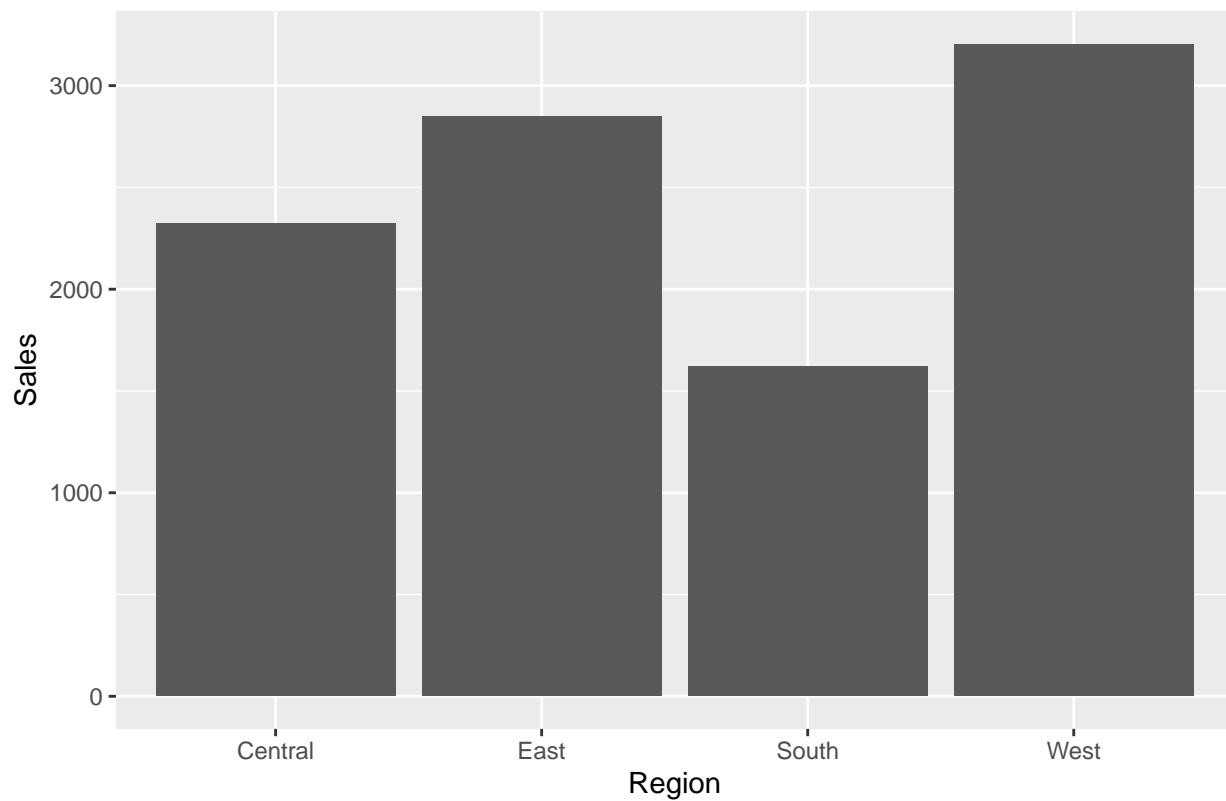
bar + coord_polar()
```



- bar charts of profits/sales by region

```
ggplot(data = data) +  
  geom_bar(mapping = aes(x = Region, fill = Sales)) +  
  ggtitle("Total Sales by region") +  
  ylab("Sales")
```

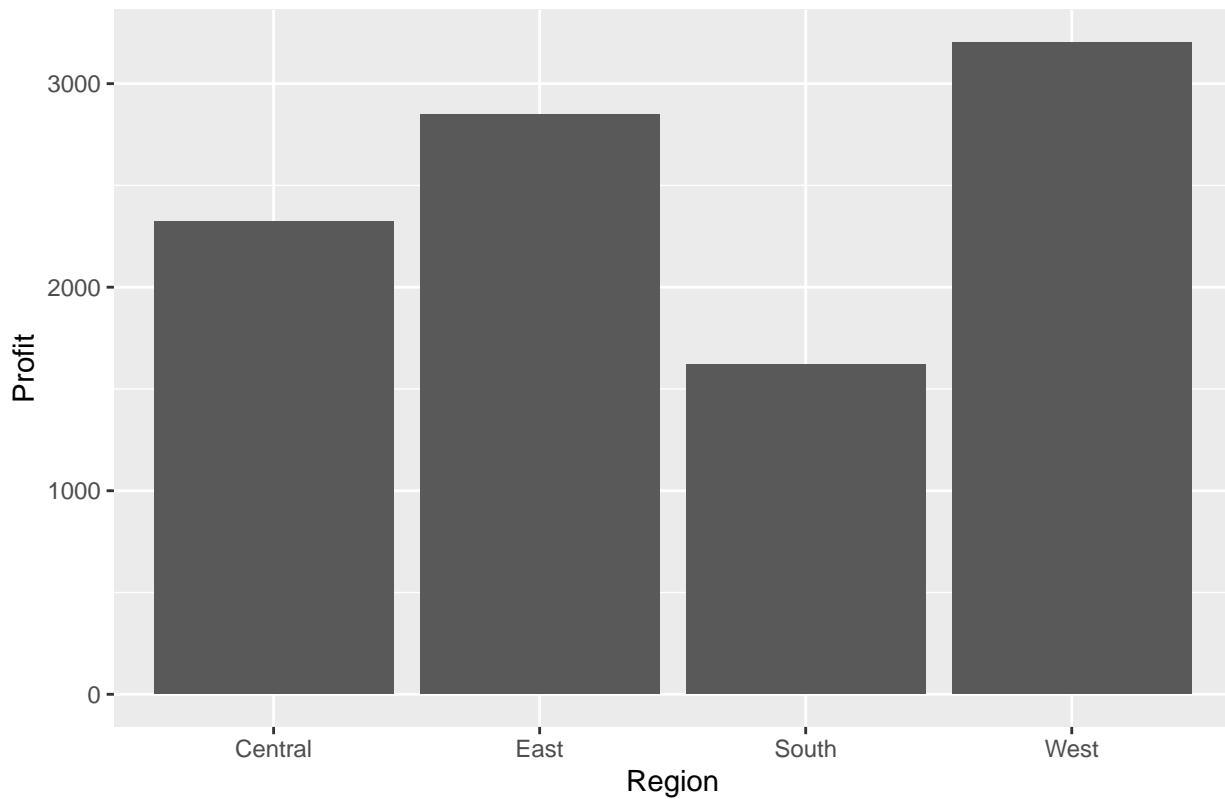
Total Sales by region



Total sales per region.

```
ggplot(data = data) +  
  geom_bar(mapping = aes(x = Region, fill = Profit)) +  
  ggtitle("Total Profit by region") +  
  ylab("Profit")
```

Total Profit by region



Look at relationship between numeric variables:

```
#subset the numeric variables:
numeric_vars<- c("Sales", "Quantity", "Discount", "Profit", "diff_in_days")
num_data <- data[numeric_vars]
```

We'll use a correlation matrix to look at the relationship between numeric variables:

```
cor(num_data)
```

```
##                  Sales   Quantity   Discount   Profit diff_in_days
## Sales      1.000000000 0.20079477 -0.0281901242 0.479064350 -0.0073535371
## Quantity    0.200794771 1.00000000  0.0086229703 0.066253189  0.0182984399
## Discount   -0.028190124 0.00862297  1.0000000000 -0.219487456  0.0004084856
## Profit      0.479064350 0.06625319 -0.2194874564 1.0000000000 -0.0046493531
## diff_in_days -0.007353537 0.01829844  0.0004084856 -0.004649353  1.0000000000
```

```
#correlation matrix with statistical significance
cor_result=rcorr(as.matrix(num_data))

cor_result$r
```

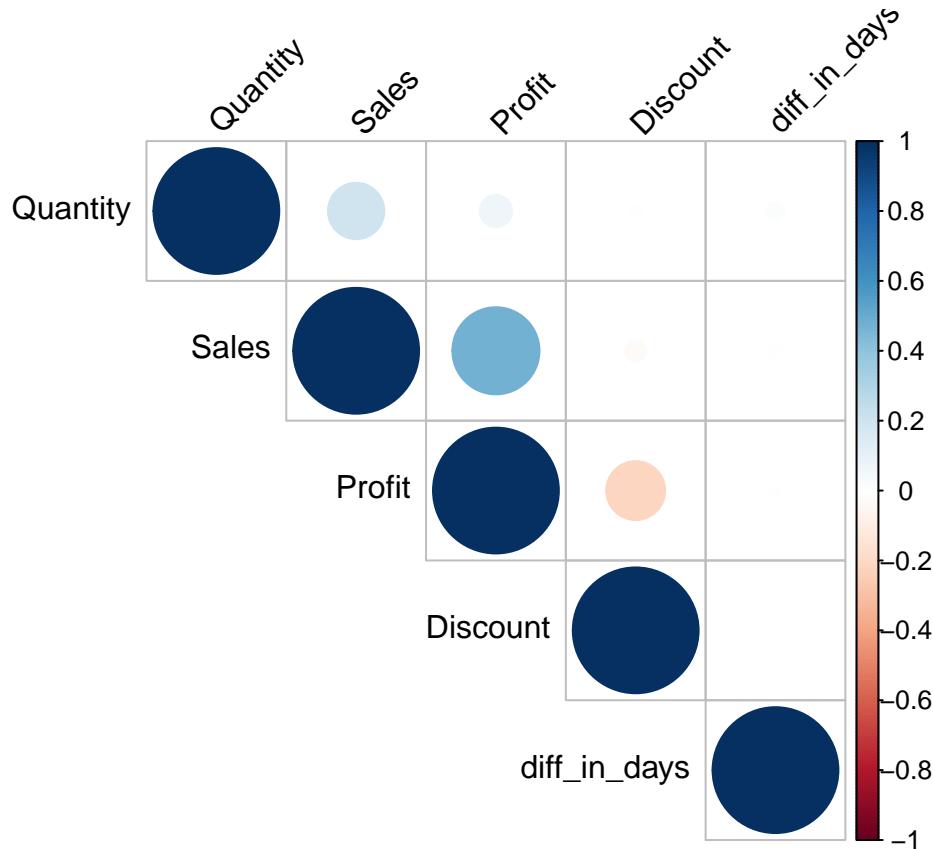
```
##                  Sales   Quantity   Discount   Profit diff_in_days
## Sales      1.000000000 0.20079477 -0.0281901242 0.479064350 -0.0073535371
## Quantity    0.200794771 1.00000000  0.0086229703 0.066253189  0.0182984399
```

```

## Discount      -0.028190124 0.00862297  1.00000000000 -0.219487456  0.0004084856
## Profit       0.479064350 0.06625319 -0.2194874564  1.0000000000 -0.0046493531
## diff_in_days -0.007353537 0.01829844  0.0004084856 -0.004649353  1.00000000000

corrplot(cor_result$r, type = "upper", order = "hclust", tl.col = "black", tl.srt = 45) #display only

```



Discount is negatively correlated with profit, whereas sales is positively correlated with profit. The time between order date and ship date (diff_in_days) is not correlated with sales, quantity, discount, or profit.

Since the difference in days between Order date and Ship date has 0 correlation with the other variables, let's drop diff_in_days for the K-means clustering analysis.

Data Preparation

```

#make a copy of the original dataset and copy to data1
data1 <- data

```

drop column Row ID because it is not necessary; it is the row number from the original excel file. The country variable is also not needed because all the values are United states. Customer_Name and Customer_ID give redundant information. So we will drop the Customer_Name column and keep only the Customer_ID column.

```

data1[,c("Row_ID","i_Row_ID", "Country", "Customer_Name")]<-NULL

```

```
head(data1)
```

```

##          Order_ID Order_Date   Ship_Date      Ship_Mode Customer_ID Segment
## 1 CA-2016-152156 2016-11-08 2016-11-11 Second Class    CG-12520 Consumer
## 2 CA-2016-152156 2016-11-08 2016-11-11 Second Class    CG-12520 Consumer
## 3 CA-2016-138688 2016-06-12 2016-06-16 Second Class    DV-13045 Corporate
## 4 US-2015-108966 2015-10-11 2015-10-18 Standard Class  SO-20335 Consumer
## 5 US-2015-108966 2015-10-11 2015-10-18 Standard Class  SO-20335 Consumer
## 6 CA-2014-115812 2014-06-09 2014-06-14 Standard Class  BH-11710 Consumer
##          City       State Postal_Code Region      Product_ID Category
## 1 Henderson Kentucky        42420 South FUR-B0-10001798 Furniture
## 2 Henderson Kentucky        42420 South FUR-CH-10000454 Furniture
## 3 Los Angeles California    90036 West OFF-LA-10000240 Office Supplies
## 4 Fort Lauderdale Florida   33311 South FUR-TA-10000577 Furniture
## 5 Fort Lauderdale Florida   33311 South OFF-ST-10000760 Office Supplies
## 6 Los Angeles California    90032 West FUR-FU-10001487 Furniture
## Sub_Category                                         Product_Name
## 1 Bookcases                                         Bush Somerset Collection Bookcase
## 2 Chairs                                             Hon Deluxe Fabric Upholstered Stacking Chairs, Rounded Back
## 3 Labels                                            Self-Adhesive Address Labels for Typewriters by Universal
## 4 Tables                                            Bretford CR4500 Series Slim Rectangular Table
## 5 Storage                                           Eldon Fold 'N Roll Cart System
## 6 Furnishings Eldon Expressions Wood and Plastic Desk Accessories, Cherry Wood
##          Sales Quantity Discount     Profit diff_in_days
## 1 261.9600        2     0.00  41.9136           3
## 2 731.9400        3     0.00  219.5820           3
## 3 14.6200         2     0.00   6.8714           4
## 4 957.5775         5     0.45 -383.0310          7
## 5 22.3680         2     0.20   2.5164           7
## 6 48.8600         7     0.00  14.1694           5

```

Model

For this K-means clustering we will use the numeric variables only: which are sales, quantity, discount, profit (columns 15 - 18). K means clustering is affected by the starting assignment points, so we will try with 25 different starting assignments (nstart = 25), and see which ones work the best.

(<https://www.datanovia.com/en/blog/k-means-clustering-visualization-in-r-step-by-step-guide/>)

```
#Compute K-means clustering with k=3 (3 initial distinct cluster centres)
set.seed(123)
```

```
results kmeans <- kmeans(scale(data1[, (15:18)]), 3, nstart =25)
```

```
#kmeans clusters to show the group of the individuals  
results kmeans$cluster
```

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



```

## [8178] 2 2 1 1 1 1 1 2 2 2 2 2 2 2 1 2 2 2 2 2 2 1 2 2 2 2 2 3 2 1 2 1 2 2 1 2 1
## [8215] 2 1 1 2 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 1 2 1 1 1
## [8252] 1 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 1 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2 2 2 1 2 2
## [8289] 2 2 2 2 2 1 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 1 2 2 2
## [8326] 2 1 1 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 1 1 2 2 2 2
## [8363] 2 2 2 2 2 2 2 2 2 2 1 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 1 2 2 2
## [8400] 2 2 2 2 2 2 2 2 2 2 1 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 1 1
## [8437] 2 2 2 2 2 2 2 1 1 1 2 2 2 2 2 2 2 2 2 2 2 1 2 2 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [8474] 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2
## [8511] 2 2 1 2 2 2 2 1 2 2 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [8548] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 1 2 2 2 2
## [8585] 2 1 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 2 2 1
## [8622] 2 2 1 2 2 2 2 2 1 2 1 1 2 1 2 2 2 2 2 2 1 1 2 2 1 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2
## [8659] 1 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2
## [8696] 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2
## [8733] 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 1 1 1 2
## [8770] 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 2 2 2 2 2 2 2 2
## [8807] 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 1 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
## [8844] 2 1 2 2 2 2 2 1 2 2 2 2 2 2 2 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 2 1 1 2 1 2 2 2 2
## [8881] 2 2 2 2 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 1 1
## [8918] 2 1 2 2 2 2 1 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2 2 2 2 1 2 2 2 2 2
## [8955] 2 2 2 1 2 1 2 2 2 2 2 1 2 2 2 2 1 2 2 2 2 2 2 1 2 1 1 1 1 1 2 1 2 2 2 3
## [8992] 2 2 1 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2
## [9029] 2 2 2 2 2 2 1 2 2 2 2 3 1 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 1
## [9066] 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2
## [9103] 2 1 1 2 1 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 1 1 1 2 2 1 1 1 2 2 2 2 2 2 2 1 2
## [9140] 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [9177] 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 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
## [9214] 2 2 2 1 2 2 2 1 2 2 2 1 1 1 2 2 2 2 2 1 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [9251] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 3 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 1
## [9288] 2 2 2 2 2 1 2 2 2 1 2 2 2 2 1 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 1 1 2
## [9325] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2
## [9362] 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 2 2 2 2 2 2 2 2
## [9399] 2 2 2 2 1 2 2 2 2 2 2 1 2 2 2 1 2 2 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [9436] 2 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2
## [9473] 2 2 2 2 1 1 2 1 1 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
## [9510] 2 2 1 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 1 2 2 2 2
## [9547] 2 2 1 2 2 2 2 2 2 2 2 1 2 2 2 2 1 2 2 2 1 2 2 2 2 2 2 1 2 2 2 2 2 2 2 1 2 2 2 2 2 2
## [9584] 2 2 2 2 2 2 2 2 1 1 1 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2
## [9621] 2 2 2 1 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 1 2 2 2
## [9658] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2
## [9695] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2
## [9732] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2
## [9769] 2 2 2 2 2 2 2 1 1 1 2 1 1 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 1 1 2 1 1 2 1 2 2 2 2
## [9806] 2 1 2 1 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2
## [9843] 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2 2 2 1 1
## [9880] 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 1 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2
## [9917] 2 2 2 2 1 1 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 2 2
## [9954] 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 2 2 2 2 2 2 2 2
## [9991] 2 2 2 2

```

```
summary(results_kmeans)
```

	Length	Class	Mode
--	--------	-------	------


```

## [9214] 2 2 2 1 2 2 1 2 2 2 1 1 1 2 2 2 2 2 1 2 2 2 2 2 2 1 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 1
## [9251] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 3 2 2 2 2 2 2 1 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 1
## [9288] 2 2 2 2 2 1 2 2 1 2 2 2 1 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 1
## [9325] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 1 2 2
## [9362] 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 2 2 2 2 2 2 2 2 2 2 2
## [9399] 2 2 2 2 1 2 2 2 2 2 2 2 1 2 2 2 2 1 2 2 2 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [9436] 2 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 2 2 1 2 2 2
## [9473] 2 2 2 2 1 1 2 1 1 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 2 2 2 2 2
## [9510] 2 2 1 2 2 2 2 1 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 1 2 2 2 2 2 2 1 2 2 2
## [9547] 2 2 1 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 1 2 2 2 2 1 2 2 2 1 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2
## [9584] 2 2 2 2 2 2 2 1 1 1 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 1 2 2
## [9621] 2 2 2 1 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 1 2 2
## [9658] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2
## [9695] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 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
## [9732] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 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
## [9769] 2 2 2 2 2 2 1 1 1 2 1 1 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 1 1 2 1 1 2 1 2 2 2 2 2
## [9806] 2 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2
## [9843] 2 2 1 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 1 1 2 2 2 2 2 2 1 1 2 2 2 2 2 2 1 1
## [9880] 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 1 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2
## [9917] 2 2 2 2 1 1 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 2 2 2 2 2 2 2 2 2 2 2
## [9954] 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 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [9991] 2 2 2 2

##
## Within cluster sum of squares by cluster:
## [1] 5207.851 16648.669 3315.203
## (between_SS / total_SS = 37.0 %)
##
## Available components:
##
## [1] "cluster"      "centers"       "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"          "iter"          "ifault"

```

<https://towardsdatascience.com/clustering-analysis-in-r-using-k-means-73eca4fb7967>

#cluster means are the centroid vectors

#clustering vector is the group that the observation is placed into

#percentage indicates compactness of the clustering or how similar observations are within the same group.

The results of this clustering indicate that the within cluster sum of squares by cluster is 37.0 % which means that the observations within a given group are not very similar to each other.

Plot K-means

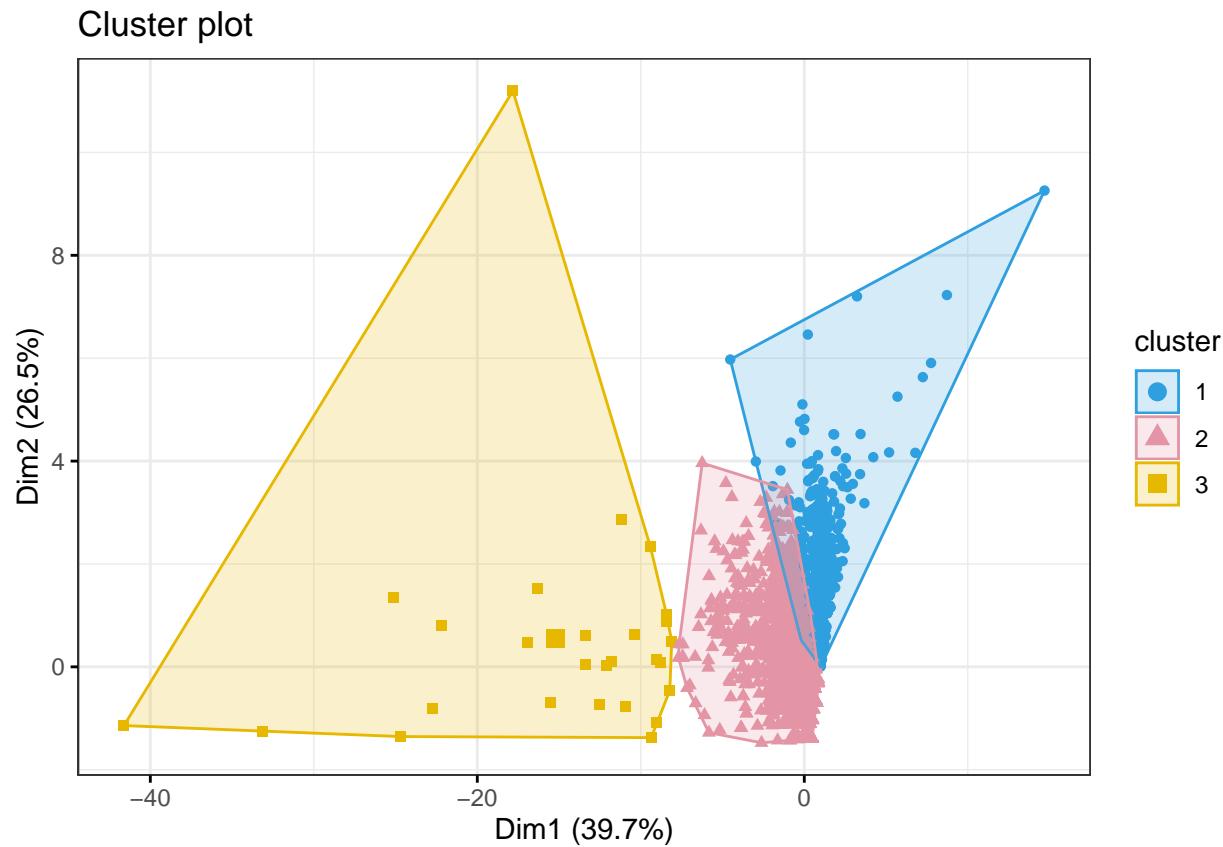
The factoextra package contains a function called `fviz_cluster()` which can be used to visualize kmeans clusters. The input required is the original dataset, and the kmeans results. These are used to produce plots which show points that represent observations.

```
fviz_cluster(results_kmeans, data = data1[, (15:18)],  
            palette = c("#2E9fdf", "#e495a5", "#e7b800"),  
            geom = "point",  
            ellipse.type = "convex",
```

```

ggtheme = theme_bw()
)

```



Reduce dimensions using Principal Component Analysis.

```

results_pca <- prcomp(data1[, (15:19)], scale=TRUE)

#Coordinates of individual observations
indiv_coordinates <- as.data.frame(get_pca_ind(results_pca)$coord)

#Add clusters obtained through the Kmeans algorithm
indiv_coordinates$cluster <- factor(results_kmeans$cluster)

#Add region from the dataset
indiv_coordinates$Region <- data1$Region

#look at the first few rows of individual coordinates
head(indiv_coordinates)

```

##	Dim.1	Dim.2	Dim.3	Dim.4	Dim.5	cluster	Region
## 1	0.04520418	-1.13666870	0.3612739	0.0807637427	0.30768311	2	South
## 2	1.15718390	-0.84275631	0.4252513	0.1851981706	0.21856052	2	South
## 3	-0.31072375	-1.08484635	-0.2274155	-0.0007838382	0.15040037	2	West
## 4	-0.63620673	2.24187253	-1.2663321	0.5805617103	1.55922466	1	South
## 5	-0.58444157	-0.08714015	-1.6947210	0.7880697439	-0.05040709	2	South
## 6	0.39080002	0.44622564	-0.7312599	-1.4739348647	-0.19665673	2	West

Percentage of variance explained by dimensions.

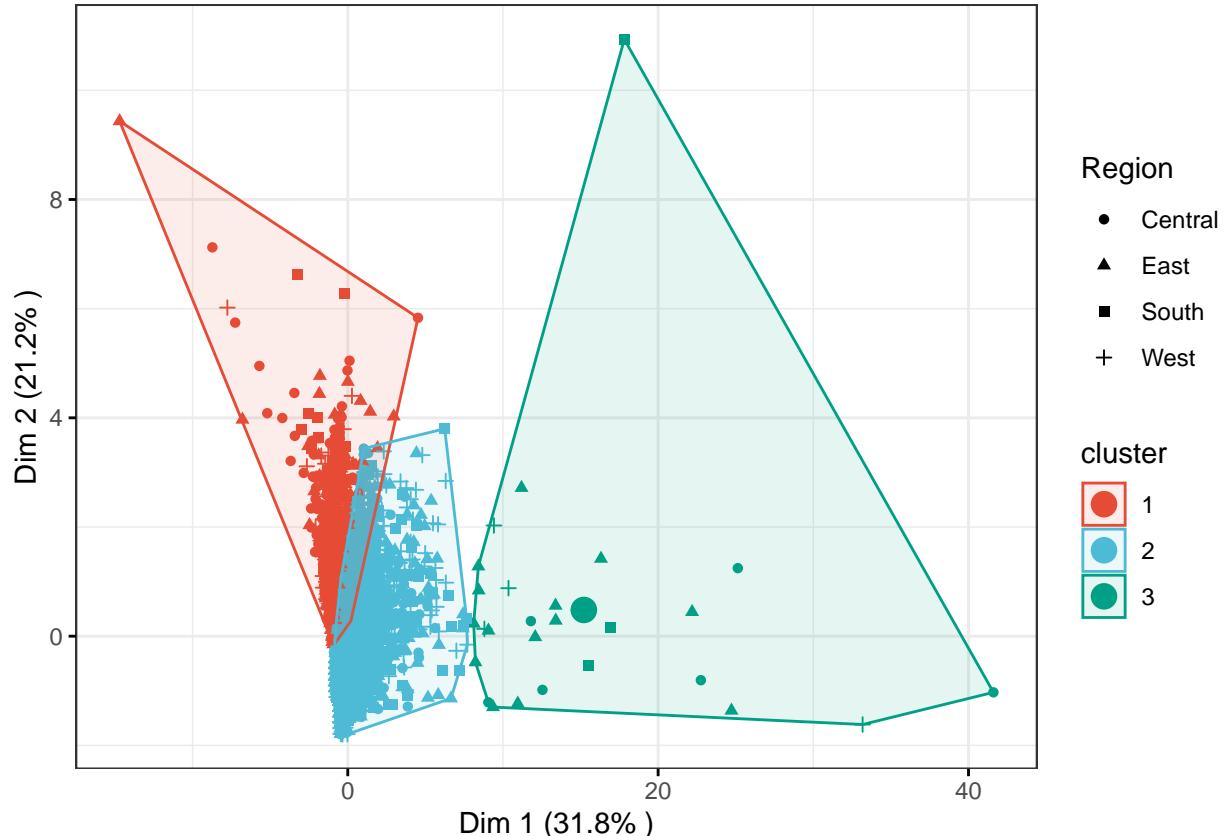
```
eigenvalue <- round(get_eigenvalue(results_pca), 1)
variance.percent <- eigenvalue$variance.percent
head(eigenvalue)
```

	eigenvalue	variance.percent	cumulative.variance.percent
## Dim.1	1.6	31.8	31.8
## Dim.2	1.1	21.2	53.0
## Dim.3	1.0	20.0	73.0
## Dim.4	0.9	17.6	90.6
## Dim.5	0.5	9.4	100.0

Variance of a group indicates how different members of a group are. Higher variance means greater dissimilarity within a group.

#To visualize the k-means clusters:

```
ggscatter(
  indiv_coordinates, x = "Dim.1", y = "Dim.2",
  color = "cluster", palette = "npg", ellipse = TRUE, ellipse.type = "convex",      #adding the concentra
  shape = "Region", size = 1.5, legend = "right", ggtheme = theme_bw(),
  xlab = paste0("Dim 1 (", variance.percent[1], "% )"),
  ylab = paste0("Dim 2 (", variance.percent[2], "% )")
) +
  stat_mean(aes(color = cluster), size = 4)      #stat_mean is used for adding the cluster centroid
```



The clustering plot shows that the groups are very close together, and overlap slightly. The clusters could be further apart with some tuning by changing the number of clusters (k).

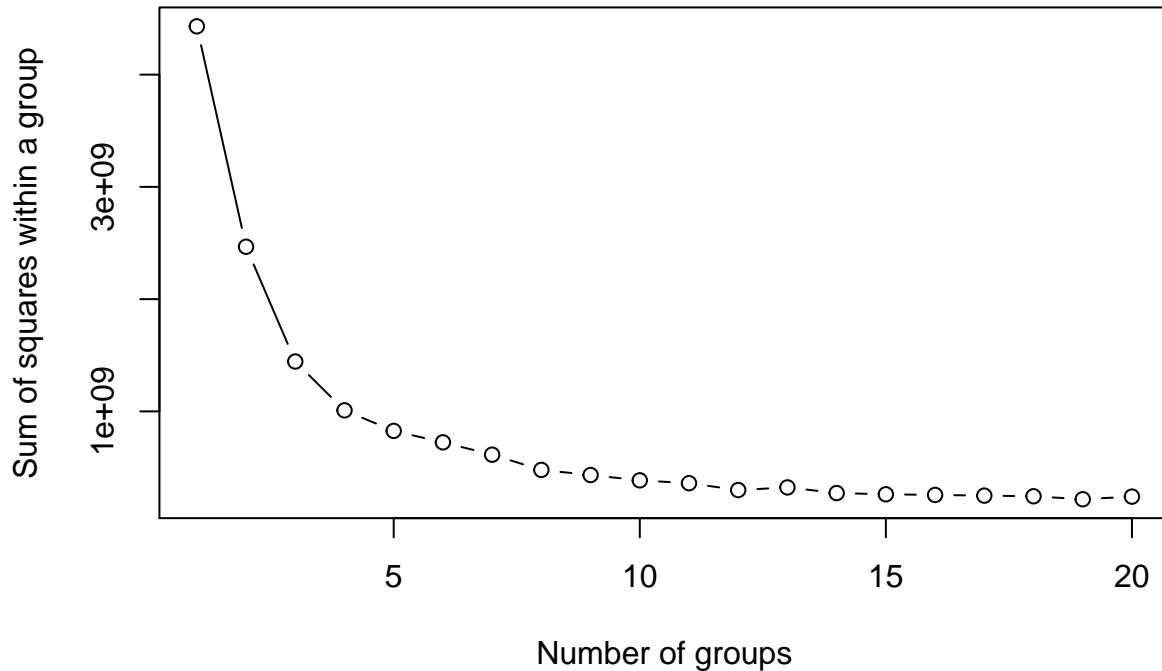
Evaluation

(<https://towardsdatascience.com/clustering-analysis-in-r-using-k-means-73eca4fb7967>)

The within sum of squares (Withinss) is a value that represents the level of dissimilarity within a group. The higher the withinss, the greater the dissimilarity within the group.

(Foncseca, 2019)

```
#To plot a within sum of squares plot for a range of different number of initial K-means centroids:  
  
#This function is from: (https://towardsdatascience.com/clustering-analysis-in-r-using-k-means-73eca4fb7967)  
  
#data is the input dataset, nc is the maximum number of initial centres  
  
wssplot <- function(data, nc=25, seed=123){  
    wss <- (nrow(data)-1)*sum(apply(data,2,var))  
    for (i in 2:nc){  
        set.seed(seed)  
        wss[i] <- sum(kmeans(data, centers=i)$withinss)}  
    plot(1:nc, wss, type="b", xlab="Number of groups",  
         ylab="Sum of squares within a group")}  
  
wssplot(data1[,15:18], nc = 20)  
  
## Warning: did not converge in 10 iterations  
## Warning: did not converge in 10 iterations
```



From the Within sum of squares plot, the optimal number of clusters is around 5. When the number of groups (k) initially increases 1 to 6, the error measures (sum of squares within a group) starts to decrease. When the number of groups is 7 or 8, the error measure starts to flatten.

The main purpose is to find a number of initial groups which achieves some fair amount of compactness (or similarity) between observations within a group. When k is too high, each cluster starts to represent individual points, whereas when k is too low, the observations may not be in the right cluster.

We can try re-running the k-means model with the number of groups, k = 4

```
set.seed(123)
clustering_results_4 <- kmeans(scale(data1[, (15:18)]), centers = 4, nstart = 25)
clustering_results_4
```

```
## K-means clustering with 4 clusters of sizes 27, 1044, 2838, 6085
##
## Cluster means:
##           Sales      Quantity     Discount      Profit
## 1  11.9621017  0.62721018 -0.4157497 11.02007172
## 2  -0.1222964  0.02095769  2.4788189 -0.58466503
## 3   0.2856822  1.19688927 -0.2981228  0.17558930
## 4  -0.1653353 -0.56459922 -0.2844025 -0.03048054
##
## Clustering vector:
## [1] 4 4 4 2 4 3 4 3 4 3 3 3 4 4 2 2 3 4 4 4 4 3 3 4 4 4 4 2 2 4 4 3 2 4 4 3 2
## [38] 3 4 4 4 4 4 4 4 4 3 3 4 4 3 3 3 3 4 4 4 4 4 4 3 4 4 3 3 4 4 4 4 4 3 4 3 4
## [75] 4 2 2 4 2 4 4 3 4 3 4 4 4 3 4 4 4 4 2 3 4 3 3 4 2 4 3 4 2 3 4 4 3 4
```

```

## [112] 4 3 3 4 4 3 4 2 4 4 3 3 3 4 2 2 3 4 4 2 4 3 4 4 4 4 3 3 4 4 4 4 3 4 3
## [149] 4 3 3 3 4 3 4 3 4 4 4 3 4 4 3 4 4 3 4 3 3 2 4 4 4 3 2 4 2 3 3 4 4 4 3 4
## [186] 4 3 4 4 3 3 3 4 3 3 4 4 4 2 4 4 2 4 2 3 4 4 4 4 3 4 4 4 4 2 2 2 4 3 4 4 3 2
## [223] 4 2 4 4 3 4 4 3 2 2 2 2 4 4 4 3 2 2 4 2 2 4 3 4 3 3 4 4 4 4 3 3 4 2 4 3 3 4
## [260] 4 3 2 2 3 4 4 4 3 4 4 3 3 4 4 4 4 4 4 3 4 2 3 3 3 3 3 2 4 3 3 3 4 2 3 4 4
## [297] 2 4 3 3 4 4 4 2 4 4 4 3 4 4 4 4 4 4 4 4 4 4 4 4 1 3 4 4 4 3 4 4 4 3 4 4 4 3 2 4 2
## [334] 2 4 4 4 4 4 4 4 2 3 3 4 4 4 4 4 4 4 4 4 4 1 3 4 4 3 3 3 4 4 4 4 4 3 4 4 4 3 3 3
## [371] 3 4 2 3 4 3 4 3 2 4 2 3 4 4 2 4 2 4 4 4 4 3 2 4 4 4 4 4 3 4 4 4 4 3 4 3 4
## [408] 3 3 4 3 4 3 4 3 4 3 3 4 4 4 4 3 4 3 4 3 4 3 2 3 3 4 3 2 3 4 3 3 4 4 4 4 4 4 4
## [445] 4 3 4 4 4 3 4 4 4 2 4 4 4 4 4 3 4 3 2 2 4 4 2 4 2 4 4 4 4 3 3 3 3 4 3 3 3
## [482] 4 4 4 4 3 3 3 4 4 4 3 4 4 2 4 3 4 4 3 2 2 4 2 4 3 4 4 4 4 1 4 4 4 4 4 3 4 4
## [519] 4 4 4 2 4 4 4 3 4 4 4 4 3 4 4 3 4 2 4 4 4 3 3 3 4 4 2 3 4 3 2 3 4 4 4 4 4 4
## [556] 4 4 3 3 4 3 3 4 4 4 4 4 3 3 4 4 3 4 3 4 4 3 4 4 4 2 4 4 4 4 2 4 4 4 4 2 4
## [593] 4 4 4 4 3 3 3 4 2 2 4 4 4 2 2 4 2 3 4 4 4 4 3 3 2 4 4 3 4 4 2 4 3 4 3 3 4 3
## [630] 4 4 3 4 4 4 4 2 4 3 4 4 4 3 4 4 4 4 3 4 4 4 3 4 2 4 4 2 3 3 3 2 3 4 4
## [667] 4 4 3 2 2 3 3 4 3 4 2 4 2 2 4 4 4 2 4 4 4 4 4 4 4 4 3 3 4 3 4 4 2 2 4 4
## [704] 4 4 3 4 4 3 3 4 4 3 4 4 3 4 4 4 3 4 4 3 3 4 4 3 2 4 4 4 4 4 3 4 4 3 4 4 3 4
## [741] 4 2 3 4 4 4 3 4 4 4 3 3 4 4 4 2 4 4 4 4 3 4 2 4 4 3 4 3 4 4 4 4 3 4 4 3 3 3
## [778] 4 4 4 4 2 4 2 4 4 4 4 4 3 4 3 4 3 4 3 3 4 3 4 4 4 3 4 4 4 4 3 4 4 4 3 4 4 4
## [815] 3 3 4 3 2 3 2 4 4 4 4 4 3 4 4 4 4 4 4 4 4 2 2 4 4 4 4 4 4 4 3 4 4 3 4 4 3 4
## [852] 3 3 4 4 4 3 4 4 4 4 4 4 3 4 4 4 4 4 4 2 2 2 4 4 3 3 4 4 4 4 2 4 3 4 4 3 4 4 3 3
## [889] 3 3 4 4 4 3 3 4 4 3 3 2 3 2 4 4 4 4 4 3 4 3 3 4 3 4 3 4 4 2 4 4 4 4 4 3 4
## [926] 4 3 4 4 4 4 4 4 3 2 4 4 4 4 4 4 4 3 3 4 3 2 3 3 3 4 3 4 4 4 4 3 4 2 4 4 4 4
## [963] 4 4 4 4 4 4 4 4 3 2 2 4 3 4 2 3 2 4 3 4 4 3 4 4 4 2 3 4 3 4 4 3 3 4 4 4 4 3
## [1000] 3 4 3 3 4 4 4 4 2 3 4 4 4 4 3 4 3 4 4 3 3 4 4 4 4 4 3 4 4 4 3 4 4 3 3 4 2 2
## [1037] 4 4 3 4 3 4 4 3 2 3 3 4 3 4 4 4 3 4 4 4 4 4 4 4 4 3 4 2 4 2 4 4 2 3 4 4
## [1074] 2 2 4 3 4 4 4 4 3 3 4 4 1 4 4 4 4 4 3 4 3 2 3 4 4 4 4 4 2 2 4 3 2 2 3 4
## [1111] 3 4 2 4 4 4 4 3 3 4 4 3 3 4 4 4 4 4 3 4 4 3 4 4 4 4 3 4 4 4 4 3 4 4 4 4 4 4
## [1148] 4 4 4 4 3 4 3 4 3 4 4 4 4 4 4 4 4 3 4 3 4 4 4 4 4 4 3 2 3 4 4 4 4 4 3 4 4 3
## [1185] 3 3 4 4 4 3 4 4 4 4 4 4 4 4 3 4 2 4 2 4 4 4 3 4 3 3 4 4 4 4 4 2 2 4 4 4 4 3 3
## [1222] 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 2 4 4 4 4 4 4 4 4 3 3 3 4 3 3 3 4 3 4 2 4 4 2
## [1259] 4 3 4 2 4 4 3 4 4 3 4 4 4 4 2 2 2 4 3 4 4 4 4 4 4 2 4 2 3 4 4 4 4 3 4 3 4 3 4
## [1296] 3 4 4 3 4 3 4 4 4 4 3 3 4 4 4 4 4 4 3 3 3 4 4 2 2 4 3 4 4 2 3 4 4 4 4 4 4 4
## [1333] 2 2 3 4 4 4 4 3 2 2 2 4 3 4 3 3 3 3 4 4 3 4 4 4 4 3 4 2 4 3 4 2 3 2 3 4 4 3
## [1370] 2 4 4 4 4 3 4 3 3 3 3 4 2 4 3 3 3 4 4 4 4 4 4 4 4 3 4 2 2 4 4 4 4 4 4 4 3 4 2
## [1407] 4 4 4 2 4 4 4 3 4 4 3 4 2 4 2 4 2 3 3 4 4 3 3 3 3 3 4 3 3 3 4 4 3 3 3 4 4 4 4
## [1444] 4 3 4 3 4 2 4 2 3 4 3 3 4 4 3 4 4 4 4 4 4 3 4 4 4 4 4 4 4 4 4 2 3 4 3 3 4 4
## [1481] 3 4 4 4 4 4 3 4 4 4 4 4 3 4 4 3 3 4 3 4 4 4 4 4 4 3 4 4 4 4 4 3 3 4 2 2 4 3
## [1518] 4 4 3 3 3 3 3 3 4 4 3 4 4 4 4 3 4 4 4 4 4 3 3 3 4 4 3 4 2 4 4 2 4 3 3
## [1555] 4 4 4 3 4 3 3 4 2 4 3 4 4 2 4 4 3 4 4 4 4 3 4 4 4 4 4 3 4 4 4 4 3 4 4 4 4 3
## [1592] 3 4 4 4 4 3 4 4 4 2 4 4 3 3 4 3 4 4 4 4 4 4 3 4 4 4 4 4 4 4 4 3 4 4 3 4 3 3 3
## [1629] 3 2 4 4 4 4 4 4 3 4 4 4 4 4 4 4 3 4 4 4 4 2 4 4 4 3 4 4 4 4 4 3 4 4 4 4 2 2 3 4
## [1666] 4 4 4 4 2 4 4 4 4 4 2 2 4 4 4 4 2 4 2 4 4 4 2 2 4 4 2 3 3 3 4 4 4 4 4 4 3 3 3 4 3
## [1703] 4 3 2 2 3 4 4 4 3 3 3 2 4 4 3 3 4 3 4 4 4 4 4 4 4 4 3 4 4 4 4 3 4 4 4 4 4 3 4 4 4
## [1740] 3 3 4 4 4 3 4 2 4 4 4 4 4 3 4 4 3 3 2 4 4 4 4 3 4 4 3 4 2 2 4 2 4 3 4 3 4 3 4 3 4
## [1777] 4 3 3 4 4 4 4 4 4 3 3 4 4 4 4 3 4 3 4 4 4 4 3 4 3 3 4 4 4 2 3 4 3 3 3 3 2 4 3 3
## [1814] 4 4 4 4 4 2 2 2 4 4 4 2 4 4 4 3 2 3 4 4 2 3 3 3 4 4 4 4 4 4 2 4 3 4 4 4 4 4 4 4
## [1851] 3 3 3 4 3 4 3 2 4 4 4 4 4 2 4 2 3 3 4 4 3 4 4 4 4 4 4 4 2 4 3 4 3 4 3 4 4 4 4 4
## [1888] 2 4 2 4 4 4 4 4 4 4 4 2 2 3 4 4 4 4 3 2 2 4 4 4 4 3 4 3 4 4 4 3 4 3 4 4 4 3 4 3 4
## [1925] 4 4 4 4 4 4 4 4 4 4 3 3 3 4 2 3 2 4 4 4 2 4 4 4 4 4 4 3 4 3 3 4 4 4 4 4 4 4 4 4 3
## [1962] 3 4 3 4 4 4 3 4 4 4 3 2 4 4 4 3 3 4 3 4 3 4 4 4 3 4 4 3 3 4 4 4 4 3 4 4 4 4 4 2 2 4
## [1999] 4 3 3 3 4 3 2 3 3 3 4 4 4 4 4 4 4 4 3 4 4 4 4 3 3 3 4 4 4 4 3 4 4 4 4 3 4 4 4 4 4 4 4
## [2036] 4 4 4 4 2 2 2 3 3 3 3 4 4 4 4 4 3 2 2 3 3 3 4 3 3 4 4 4 4 3 4 3 4 4 4 4 4 4 2 3
## [2073] 4 4 3 4 3 3 3 3 4 4 4 3 4 4 4 4 4 3 4 3 3 4 4 4 4 4 3 4 4 4 4 4 4 4 2 4 4
```

```

## [2110] 4 3 4 4 3 4 4 4 3 2 4 4 3 3 3 4 4 4 4 3 3 3 4 2 3 2 2 2 4 3 4 2 3 4 3 4
## [2147] 2 3 4 3 4 3 4 4 2 3 4 4 4 2 4 4 4 4 2 4 2 4 4 2 4 4 2 4 4 3 3 3 3 4 4 3
## [2184] 3 3 4 4 3 4 3 4 4 4 4 4 3 4 4 4 4 3 3 4 4 4 4 2 4 4 3 3 3 3 4 4 3 4 4 4 4
## [2221] 3 2 3 4 4 2 4 4 4 4 4 3 3 4 3 4 3 3 2 4 3 2 4 4 4 4 4 3 3 3 3 4 4 4 4 4 4
## [2258] 3 4 4 4 4 4 4 4 4 4 4 4 3 3 3 4 4 4 4 3 4 4 4 3 4 3 4 3 4 4 3 3 4 4 3 4 4
## [2295] 4 4 3 4 3 4 4 4 4 4 3 4 4 4 4 3 4 4 4 4 3 4 4 4 3 4 3 4 4 3 4 4 3 4 4 4 4
## [2332] 3 3 4 3 4 3 4 4 4 3 3 4 4 3 3 2 3 3 4 4 4 3 3 3 3 4 4 3 2 2 3 4 2 4 4 4 4
## [2369] 3 4 4 2 4 3 4 4 2 4 3 4 4 3 4 3 4 4 4 4 4 3 3 4 4 4 4 3 4 3 4 3 3 3 3 3 3
## [2406] 3 3 3 4 3 4 3 4 4 3 4 3 4 3 4 4 4 4 4 3 2 2 3 2 2 2 3 4 3 4 4 4 3 2 4 4 3 4
## [2443] 3 3 3 4 4 3 4 3 3 4 4 3 3 4 3 4 4 4 4 4 3 4 4 4 4 3 4 3 4 4 4 4 3 3 3 3 4 4
## [2480] 4 4 3 4 3 3 4 4 4 4 3 4 3 1 3 3 3 4 4 4 4 4 4 2 2 4 4 1 3 2 4 3 3 3 2 4 4 4
## [2517] 3 4 3 3 4 4 4 4 4 3 4 2 2 4 4 3 4 3 4 4 2 4 3 4 3 4 4 4 4 4 3 3 4 3 4 4 4 3 4
## [2554] 4 3 4 2 4 3 3 4 4 4 4 4 4 4 3 3 3 4 4 4 4 4 3 3 4 3 4 4 4 4 3 4 4 4 3 4 4 4 4
## [2591] 2 2 4 3 4 4 4 4 4 4 3 4 4 4 4 2 4 4 4 4 3 4 4 2 4 4 3 4 3 4 4 4 4 3 4 1 3 3 4
## [2628] 4 2 2 4 4 4 4 4 2 4 4 4 4 3 4 3 4 3 4 4 4 3 4 3 4 4 4 4 4 4 3 4 3 3 4 4 4
## [2665] 2 2 4 4 4 4 4 3 4 3 4 3 4 2 2 3 3 4 4 4 3 4 2 2 3 4 2 4 4 3 4 4 2 1 4 4 4
## [2702] 4 4 4 3 4 4 4 3 3 2 4 3 4 3 4 4 4 4 4 2 4 3 3 3 4 2 2 2 3 4 3 4 4 4 4 4 3 4
## [2739] 4 4 4 3 4 3 4 4 4 4 2 4 3 4 4 4 3 3 4 4 4 4 3 4 3 2 2 4 4 3 2 3 4 3 4 4 4
## [2776] 4 3 4 3 4 4 3 4 4 2 4 4 2 3 2 4 4 4 3 4 3 4 4 4 3 3 4 4 4 4 3 4 4 4 3 4 3 3
## [2813] 4 3 4 3 4 4 3 3 4 4 4 4 4 3 4 2 4 3 3 4 3 4 3 4 4 4 4 3 2 4 3 3 4 4 3 4 2 3 3
## [2850] 2 4 4 4 4 4 3 3 4 4 4 4 3 3 4 4 3 4 4 3 4 4 4 4 4 3 4 4 4 4 3 4 4 4 3 4 2 4 4 3
## [2887] 4 4 3 3 4 4 4 3 3 4 3 3 4 4 4 3 3 3 4 4 4 4 3 4 4 4 4 3 4 4 4 4 4 4 3 4 3 4 2
## [2924] 4 4 4 4 4 2 4 4 4 4 4 4 4 3 3 3 4 4 3 4 3 4 3 4 3 4 3 4 4 4 4 4 4 4 2 4 4
## [2961] 4 3 4 4 3 4 2 4 4 3 2 4 4 3 4 3 4 3 4 4 4 4 4 2 4 4 3 4 4 3 3 4 3 4 3 4 3 4
## [2998] 4 4 2 4 2 3 4 4 4 3 4 4 4 3 2 2 2 2 4 2 3 2 3 4 4 4 3 4 3 4 4 4 4 3 4 4 4 4 4
## [3035] 3 3 4 3 4 4 3 4 4 3 2 3 4 3 4 4 4 4 4 4 3 4 3 4 4 4 3 3 3 3 3 3 4 3
## [3072] 4 3 4 4 4 2 4 3 4 4 4 4 4 2 4 2 2 3 3 3 3 4 4 4 3 2 4 4 4 4 3 4 2 3 4 3 4
## [3109] 4 4 3 4 4 3 3 3 3 4 3 4 4 4 4 2 2 4 4 4 4 4 4 3 4 4 4 4 2 4 4 4 4 2 4 2 3 3 3
## [3146] 3 4 3 3 4 3 2 4 3 4 4 2 4 3 4 2 4 3 4 3 4 4 3 3 3 2 3 4 3 2 4 4 4 4 3 4
## [3183] 4 3 4 2 4 3 4 4 4 4 2 2 3 4 4 2 4 3 4 4 3 4 4 3 3 3 3 3 4 2 3 4 4 3 4 4 4
## [3220] 3 4 4 4 4 4 3 4 4 4 4 4 3 2 2 4 4 4 4 4 3 3 2 3 2 3 4 4 3 4 4 4 4 3 4 4 4 4
## [3257] 4 2 2 2 4 4 3 4 4 4 4 4 4 4 4 3 3 3 4 3 3 4 4 4 4 2 4 4 4 4 3 3 4 4 4 4
## [3294] 4 2 3 2 4 3 2 4 3 4 2 4 4 4 2 4 3 3 3 4 4 4 4 2 3 4 3 4 3 4 4 2 2 4 3 3 4
## [3331] 4 4 4 2 4 4 4 4 3 4 3 4 4 4 3 4 3 3 2 4 4 4 4 4 4 4 4 3 2 4 4 4 4 4 2 4 4
## [3368] 4 4 4 3 3 4 4 3 4 4 3 2 4 3 4 4 3 3 4 4 4 4 2 2 4 3 4 2 3 4 4 4 4 3 4 4 2 2
## [3405] 4 4 4 2 4 4 3 4 4 4 4 4 4 4 3 4 4 4 4 3 3 4 4 4 4 3 4 2 3 4 4 3 4 3 2 2 2 4 3 4
## [3442] 3 4 3 4 3 4 4 4 3 4 4 4 4 4 4 4 4 4 2 2 4 4 3 4 4 4 4 3 3 4 4 4 4 4 4 3 3 3
## [3479] 4 3 3 4 3 4 4 4 4 4 4 3 3 4 4 4 3 3 4 4 3 4 4 3 4 4 2 4 4 4 4 2 3 3 4 4 2 3 3 4 4
## [3516] 4 3 4 4 2 4 3 4 3 3 4 4 3 4 4 3 3 4 4 3 4 4 4 4 4 3 3 2 3 4 4 3 2 3 4 4 3 4 3
## [3553] 3 2 4 3 3 4 4 4 4 2 2 4 4 4 3 2 4 3 3 4 4 4 4 3 4 4 4 4 3 3 4 4 4 4 4 2 4
## [3590] 4 3 4 4 2 3 4 4 3 4 4 3 4 3 3 4 4 4 4 4 4 4 4 4 4 4 3 4 2 4 4 3 4 3 3 4 3 4
## [3627] 4 4 3 4 3 4 3 4 3 4 4 4 3 4 3 2 4 4 4 4 3 4 4 2 3 3 3 4 4 3 4 4 4 4 4 3 4 3
## [3664] 4 3 2 4 3 3 3 4 4 4 2 4 4 4 4 4 4 4 4 4 3 3 3 4 2 3 3 3 2 4 3 4 3 3 4 4 3 4
## [3701] 4 4 3 4 4 3 4 2 2 3 4 3 4 4 4 3 3 4 3 4 4 3 2 4 4 4 4 4 4 3 4 3 4 4 4 4 4 4
## [3738] 4 4 4 4 4 3 4 4 4 4 4 3 4 4 4 4 4 3 3 3 3 4 4 4 4 3 4 4 2 4 4 4 3 3 4 3 4 4
## [3775] 4 4 3 3 4 4 4 4 4 4 4 2 3 4 4 3 3 2 3 4 4 4 4 4 4 2 3 4 4 4 4 4 4 4 3 2 2 3 4
## [3812] 4 4 3 3 4 4 4 4 3 4 4 4 3 3 4 3 3 3 4 4 4 4 4 4 2 4 4 4 4 4 4 3 4 4 4 4 4 3 3 4
## [3849] 3 3 3 4 2 4 3 2 3 4 3 2 4 4 4 4 2 4 3 3 4 3 4 4 4 4 4 2 4 4 4 2 4 4 4 4 4 3 4 4
## [3886] 2 4 4 4 4 4 4 3 3 3 2 3 4 4 4 4 4 4 3 3 4 2 4 2 4 4 4 2 4 4 4 2 4 4 4 2 4 4 4 4
## [3923] 3 3 4 4 4 3 3 4 3 4 4 3 3 4 4 4 4 4 3 4 3 4 4 4 4 4 4 3 4 4 4 4 4 4 3 4 4 4 4 4 2 4
## [3960] 4 3 3 4 4 4 4 4 4 2 2 3 3 4 4 4 3 3 3 4 4 4 4 4 4 4 4 3 4 4 3 3 3 4 4 4 4 4 4 3 4 3
## [3997] 3 4 3 3 3 3 4 2 4 4 4 3 4 3 4 4 4 4 4 3 3 4 3 4 4 2 3 3 3 4 4 3 4 2 4 4 4 3 3 3
## [4034] 4 2 3 4 3 4 4 4 3 3 3 2 2 4 3 4 4 4 2 3 4 3 4 3 3 4 4 4 4 3 4 3 4 3 4 3 3 3 2
## [4071] 2 4 4 3 4 4 4 4 4 4 3 2 2 4 2 4 4 4 3 4 4 4 4 3 3 1 4 2 2 3 4 3 4 2
```

```

## [4108] 4 2 4 4 4 4 4 4 3 4 4 3 3 4 3 4 4 4 4 3 3 2 4 2 4 4 3 4 3 3 3 2 4 3 4 3 3
## [4145] 4 4 4 2 4 4 4 3 4 4 2 2 4 2 4 4 4 4 4 3 3 4 4 4 3 4 3 4 4 4 4 3 4 3 3 2
## [4182] 4 4 4 4 3 2 2 4 4 1 3 4 4 4 3 4 4 2 4 4 3 3 3 4 4 4 4 3 4 4 4 4 4 4 4 4 3
## [4219] 3 4 3 3 4 4 2 2 4 4 4 2 2 4 3 2 4 3 2 3 3 4 4 4 4 4 4 4 3 3 3 3 3 3 3 4 4 4
## [4256] 2 2 2 4 4 3 4 4 2 3 2 3 2 3 3 3 4 3 4 4 2 4 1 4 3 4 4 4 4 4 4 2 4 4 3
## [4293] 4 2 2 4 4 3 4 3 4 3 4 3 4 3 4 4 3 4 4 4 4 4 4 4 4 4 3 4 4 2 2 2 4 4 4 3 3
## [4330] 4 3 4 3 3 4 4 4 4 4 4 4 4 4 3 3 3 4 4 4 4 4 2 3 4 4 4 4 2 2 4 4 2 4 4 4 4 3 4
## [4367] 3 4 3 3 4 3 4 3 3 3 4 3 4 4 4 4 4 2 4 4 3 4 4 4 4 3 4 2 4 4 4 3 4 3 4 3 4
## [4404] 4 4 4 4 4 4 4 3 4 4 4 4 4 3 3 4 2 3 4 4 4 3 3 4 3 4 4 4 3 4 4 4 3 4 2 3 2 4 4
## [4441] 4 3 3 3 3 3 4 3 4 4 4 3 4 3 4 4 4 3 4 4 4 4 2 4 4 4 3 4 4 4 4 4 4 2 4 4 4 4
## [4478] 4 4 3 3 4 3 4 4 4 4 2 3 2 4 4 4 3 3 3 3 2 4 4 3 3 3 3 4 4 4 4 4 4 4 4 4 2 4
## [4515] 4 4 3 3 4 3 4 4 4 3 3 4 3 4 4 2 4 4 4 4 4 4 4 4 3 2 4 3 4 3 2 4 2 4 4 4
## [4552] 4 4 4 4 4 4 2 4 4 4 4 3 4 3 4 2 4 2 4 4 4 4 4 3 4 3 4 4 4 4 3 3 4 3 4 3 3 3 4
## [4589] 4 3 4 4 4 4 4 4 4 3 4 3 4 3 4 4 4 4 4 3 4 4 4 3 3 2 3 4 4 4 4 3 4 3 4 3 4 4 4
## [4626] 4 3 4 3 4 4 3 4 3 4 4 4 4 4 3 4 2 4 4 4 4 4 3 3 3 4 4 3 4 4 4 4 4 3 4 3
## [4663] 4 4 2 4 4 4 2 3 3 4 4 4 4 4 4 4 3 4 4 3 3 2 2 2 3 3 2 4 4 4 3 4 4 4 4 4 3 4
## [4700] 4 4 4 2 2 3 4 4 3 4 4 4 2 3 2 4 4 4 4 3 3 4 2 4 4 2 4 3 4 3 4 4 4 4 4 4 4 4
## [4737] 4 4 4 4 4 4 4 4 4 4 4 4 3 4 4 4 4 4 4 2 4 2 2 3 4 4 4 4 4 4 4 3 4 4 4 4 3 2
## [4774] 3 4 4 4 4 3 4 4 2 3 3 3 2 4 4 3 3 2 4 4 4 4 2 3 4 3 4 2 2 4 3 4 3 4 4 4 4
## [4811] 4 4 3 4 4 4 4 4 4 4 2 3 2 2 4 4 4 4 4 4 3 4 4 4 3 4 4 2 3 2 4 2 3 4 4 4 3
## [4848] 3 4 4 3 3 3 4 4 4 4 4 4 3 3 4 3 4 4 4 3 3 4 4 4 3 4 2 4 2 3 4 4 2 4 3 3 4 3
## [4885] 4 3 4 3 4 3 4 4 4 4 4 4 3 4 4 2 4 4 3 4 4 4 3 2 4 4 3 4 2 2 4 4 3 4 4 4 4 2
## [4922] 4 4 3 4 4 2 4 4 3 4 4 4 2 4 4 3 3 4 3 3 4 4 4 4 4 4 4 4 3 4 4 4 4 4 3 4 4 2
## [4959] 3 4 4 4 4 4 2 4 4 4 4 4 4 4 4 4 3 3 4 3 4 2 4 4 4 4 3 3 4 3 2 4 4 2 3 3 3
## [4996] 4 4 2 4 3 4 3 4 4 4 3 3 4 2 3 2 2 2 3 3 3 2 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3
## [5033] 4 4 3 4 4 3 4 4 3 4 4 3 3 3 4 3 2 4 4 4 4 3 4 3 3 3 2 4 3 4 4 4 4 4 4 2 3 4
## [5070] 4 4 3 4 4 4 3 3 4 4 4 4 4 4 4 4 2 3 4 2 4 4 4 4 2 2 4 2 2 4 4 4 4 4 4 3 4 4
## [5107] 4 3 3 3 3 4 2 4 4 4 4 4 4 3 3 3 4 4 4 4 3 3 3 4 4 4 3 3 4 3 4 4 4 4 3 4 3 3
## [5144] 4 4 4 4 4 4 4 4 4 3 4 4 4 4 4 4 2 4 4 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 2 4 4 4 4
## [5181] 4 3 4 4 3 3 3 4 3 3 4 4 3 3 4 4 4 3 3 4 4 4 4 4 4 4 3 4 4 4 4 4 4 4 3 4 4 3 3 4
## [5218] 4 3 3 4 3 3 3 4 3 3 4 4 4 3 3 4 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 2 4 4 4 4 2 4 4 3 4 4 4
## [5255] 4 4 4 3 3 4 4 2 4 4 4 4 4 4 4 3 4 3 2 4 4 3 3 4 4 4 4 4 4 4 4 3 2 2 4 4 3
## [5292] 3 4 4 3 4 4 3 4 3 3 4 2 3 4 4 4 3 2 4 4 4 4 3 4 4 4 4 4 2 2 2 4 4 3 3 3 4 4
## [5329] 4 4 4 3 4 4 4 4 4 3 3 3 4 3 4 3 4 3 4 4 4 3 3 3 4 3 4 4 4 4 4 3 3 3 3 4 4 4 3 3 3 4 4
## [5366] 4 4 4 4 4 4 4 4 3 4 3 4 3 4 4 3 3 3 4 3 3 4 4 4 4 4 4 4 3 4 3 3 3 3 3 2 3 4 4
## [5403] 4 4 3 4 4 4 4 4 4 3 4 3 2 4 4 4 4 3 4 2 4 4 3 4 3 3 3 4 4 4 4 4 4 2 4 3 4 4 2 4
## [5440] 4 4 4 4 3 4 4 3 3 4 3 4 4 3 3 4 4 3 4 3 3 4 4 2 4 3 2 4 4 4 4 4 3 3 4 2 4 4
## [5477] 3 4 4 2 4 4 4 4 4 2 4 4 4 4 4 4 4 4 2 4 4 4 4 4 4 3 4 4 4 4 4 4 2 4 4 4 3 4 4 4 4 4
## [5514] 4 4 4 3 3 4 4 3 4 3 4 4 4 4 3 4 3 3 4 4 4 4 4 4 4 4 4 2 4 2 2 2 3 4 4 4 4 3
## [5551] 4 3 4 3 4 4 4 4 4 3 4 4 4 4 4 3 4 4 4 4 4 4 2 4 4 4 4 3 4 4 3 3 4 3 4 4 4 4 4
## [5588] 3 4 4 4 4 4 2 4 4 4 4 4 4 3 4 4 3 3 4 4 4 3 4 4 4 4 4 3 4 3 3 3 4 4 3 4 4 3 4 3 4
## [5625] 4 4 3 3 4 3 4 3 4 3 3 4 4 3 4 3 4 4 4 3 3 3 3 4 3 3 4 3 4 4 2 4 4 4 2 4 4 2 4 4
## [5662] 4 3 4 2 4 4 4 3 4 4 4 4 4 4 3 4 2 3 4 4 4 4 3 3 3 4 3 3 3 4 3 3 4 3 4 2 2 4 2 2
## [5699] 4 3 4 3 4 4 4 4 4 4 4 4 4 3 4 4 4 3 3 3 4 3 4 3 4 4 4 4 4 4 3 3 3 4 4 4 4 4 3 4 4 4
## [5736] 4 4 4 4 4 4 4 4 4 3 3 4 4 4 4 3 2 4 3 3 4 2 3 4 2 4 4 4 4 4 3 3 3 4 3 3 4 4 4 4 3
## [5773] 4 4 3 4 4 4 4 4 4 4 4 4 2 3 2 3 3 2 3 4 4 4 4 4 4 4 4 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4
## [5810] 4 4 4 4 4 4 4 4 4 3 4 4 4 4 3 4 4 4 4 4 4 4 4 4 4 3 2 2 3 4 3 3 3 4 4 3 4 3 3 3
## [5847] 4 4 4 4 4 4 4 4 3 4 2 2 4 4 4 4 4 4 4 4 3 4 2 3 3 4 4 4 3 4 4 3 4 3 4 3 4 4 3 3 3
## [5884] 4 3 4 4 4 4 4 4 4 4 4 4 3 4 4 4 3 4 4 4 3 4 4 4 4 4 4 4 4 4 4 4 3 3 3 4 4 3 3 3 4
## [5921] 4 4 4 2 3 3 3 4 4 2 4 2 4 4 4 3 4 4 4 4 3 4 4 4 4 3 4 4 4 4 3 4 4 4 3 4 3 4 3 4 2 3
## [5958] 3 4 4 4 3 4 4 4 4 3 4 4 4 4 4 4 4 4 3 4 4 3 3 4 4 4 4 4 4 3 3 4 4 3 4 3 4 3 3 3 4 3
## [5995] 4 3 4 2 4 4 4 4 2 2 4 4 4 4 4 3 4 4 3 3 4 3 4 3 4 4 3 4 4 4 4 3 4 3 4 3 3 4 4 4
## [6032] 4 3 4 2 2 4 3 4 4 4 4 4 3 4 4 2 2 3 3 3 4 4 2 2 3 4 3 4 4 4 3 4 4 4 3 4 3 4 3 3 3
## [6069] 4 4 4 4 4 4 4 4 3 4 4 4 3 2 2 3 4 4 2 4 4 3 4 4 4 4 4 4 3 4 3 4 3 4 4 2
```

```

## [6106] 4 2 3 4 2 3 4 3 3 4 3 3 4 4 4 4 3 2 4 4 3 4 4 4 4 3 3 4 4 4 2 3 4 4 3 4
## [6143] 4 3 4 4 4 4 4 3 3 3 4 4 4 4 4 4 4 3 3 4 4 4 2 4 4 4 4 4 4 4 3 4 4 3 3 4
## [6180] 4 3 4 4 3 3 3 4 3 3 4 3 4 3 2 3 2 4 3 4 4 3 4 3 3 4 3 4 4 4 3 3 4 4 4 4
## [6217] 3 4 3 4 3 3 4 3 4 2 4 4 2 4 4 4 4 3 3 3 4 4 4 4 4 4 3 2 4 3 4 4 4 4 4 2 3
## [6254] 4 3 4 2 4 4 4 4 4 2 4 3 4 3 4 3 4 3 3 3 4 4 3 3 4 4 4 4 4 3 3 2 4 4 3 4 4 4 4 2 3
## [6291] 3 4 4 4 4 3 3 4 3 4 4 2 4 3 4 3 4 3 4 4 4 4 3 4 4 4 4 4 4 4 3 3 4 4 4 4 4 3 4 4 4
## [6328] 4 3 4 4 4 4 3 4 3 4 3 4 4 1 4 4 4 4 4 4 3 4 4 4 2 4 2 4 4 3 3 4 4 4 3 4 3 3 4
## [6365] 3 4 4 4 3 2 4 4 4 4 4 3 4 4 4 3 3 2 2 2 4 3 2 4 4 3 4 4 4 4 3 4 4 4 4 3 3 3 3
## [6402] 4 4 4 4 4 3 3 3 4 4 4 4 4 3 2 4 2 4 4 4 4 3 3 4 1 2 4 2 4 2 2 4 3 3 4 3 4
## [6439] 4 3 3 4 4 3 4 3 2 4 4 4 4 4 4 4 3 4 3 4 3 3 4 4 3 4 3 3 3 3 4 3 3 3 4 3 3 4 2
## [6476] 2 4 4 4 3 3 4 4 2 3 4 3 2 3 3 4 4 3 4 4 4 4 4 3 3 3 3 3 2 4 4 4 4 4 3 4 4 3 4 3
## [6513] 4 3 4 4 3 3 4 4 1 4 3 4 4 2 2 3 4 4 2 3 4 3 3 3 2 4 4 3 3 4 4 4 2 4 4 4 3 4
## [6550] 3 4 4 3 4 4 4 2 4 3 3 4 2 4 4 4 3 2 3 4 4 4 4 4 4 3 3 4 3 3 4 4 4 4 4 4 4 3 4 3
## [6587] 3 4 4 4 4 4 3 4 3 2 4 2 4 4 4 4 3 3 4 4 3 4 3 4 3 4 4 4 3 4 4 3 4 3 3 3 4 3 3 4 3
## [6624] 3 4 4 1 4 3 2 4 3 4 4 2 4 3 4 3 2 3 4 3 4 3 4 2 4 2 4 3 2 4 3 4 4 4 4 3 4 2
## [6661] 4 4 3 3 3 3 4 4 3 4 4 3 4 3 4 3 4 3 4 4 4 4 3 4 2 3 3 4 3 4 3 3 4 4 4 4 4
## [6698] 4 4 4 4 4 2 4 4 4 3 4 3 4 3 4 2 2 4 3 3 4 4 4 3 4 3 4 4 4 4 3 3 2 4 4 4 4 4
## [6735] 4 4 4 3 4 4 4 4 4 3 3 3 3 4 4 4 4 4 4 4 4 3 4 3 4 4 4 4 4 3 4 4 4 4 2 3 3 3
## [6772] 2 4 4 3 4 4 3 4 4 4 3 4 4 3 4 4 3 4 3 4 4 4 4 2 2 2 2 2 2 4 4 4 4 4 4 4 4
## [6809] 2 2 2 4 2 3 2 4 3 3 2 2 4 2 3 3 4 4 1 4 3 4 4 4 4 4 4 3 4 4 3 3 2 4 4 3 3
## [6846] 4 4 4 4 4 3 3 4 4 4 4 4 4 4 4 4 3 3 3 4 4 4 4 3 4 3 4 4 3 4 2 4 4 3 2 4 3 3 3
## [6883] 4 3 3 4 4 4 4 3 3 4 4 4 4 4 3 2 3 2 3 3 4 2 3 4 4 4 4 2 4 4 4 4 2 4 4 3 4
## [6920] 2 3 4 3 4 3 3 4 4 4 3 4 4 4 4 4 4 4 4 2 4 4 4 4 3 3 4 3 4 4 4 4 4 3 2 4 4 4 3 4
## [6957] 3 4 4 4 4 4 3 4 3 4 4 3 4 3 4 2 4 4 4 4 4 3 3 4 4 3 4 4 4 4 3 3 2 4 3 4
## [6994] 3 3 4 2 4 4 4 4 4 3 4 4 4 4 2 2 4 4 3 4 4 4 4 4 4 3 4 4 4 4 3 3 3 3 4 4 3 4
## [7031] 4 4 4 4 4 4 4 4 4 4 4 4 4 3 4 4 4 4 4 4 4 3 4 4 4 4 3 4 3 4 4 4 4 4 4 4 4 3 3
## [7068] 2 3 4 4 4 4 3 2 4 3 2 3 4 2 4 2 4 3 2 4 4 4 4 4 4 4 3 3 3 4 3 3 3 4 4 2 4
## [7105] 3 4 3 4 4 4 4 4 3 4 3 4 4 4 4 4 4 2 3 4 4 4 4 4 4 3 4 4 3 4 4 4 4 4 2 4 2 4 4
## [7142] 3 3 2 2 2 4 4 4 4 4 2 4 2 3 3 4 4 4 4 4 4 4 3 3 4 4 4 4 4 3 4 4 4 4 3 3 4 2 4 2
## [7179] 2 4 4 4 3 3 4 4 4 4 2 2 4 4 2 3 3 4 4 3 4 4 4 4 4 2 3 2 4 4 3 4 4 4 4 3 4 4 3
## [7216] 4 4 4 4 4 2 4 4 4 3 4 3 3 3 4 4 4 4 4 3 4 4 4 4 2 3 3 3 3 4 3 3 3 4 2 3 4 4 4
## [7253] 4 4 3 4 4 3 3 3 4 3 2 4 3 4 2 4 4 3 4 4 4 2 3 3 3 4 4 3 3 2 2 3 4 4 4 4 2
## [7290] 3 4 3 2 3 4 4 3 3 2 2 3 2 3 4 4 4 4 4 4 3 2 2 3 4 4 4 4 2 3 3 3 4 4 4 4 3
## [7327] 4 4 3 3 2 4 2 4 4 4 4 4 4 4 4 3 4 3 2 4 4 4 4 2 3 3 3 3 3 3 4 4 3 2 4 4 4
## [7364] 2 3 2 4 3 2 4 4 3 4 3 4 4 4 4 3 4 3 3 4 4 4 4 4 4 3 3 4 4 3 4 3 2 4 3 3 4
## [7401] 4 3 4 3 4 4 4 3 3 4 4 4 4 4 4 4 4 3 2 3 2 3 2 4 2 4 4 4 2 4 4 4 4 4 2 4 3 3
## [7438] 3 3 3 4 4 4 4 4 4 2 3 3 4 3 2 3 4 4 4 4 4 4 3 3 3 4 3 4 3 4 3 4 2 4 3 4
## [7475] 3 4 4 4 3 4 4 4 4 3 4 3 3 3 4 3 3 4 3 2 4 4 4 4 4 4 4 4 4 4 2 4 4 4 4 3 3
## [7512] 3 3 3 4 3 4 3 4 4 4 2 2 4 3 3 3 4 3 4 4 4 4 4 4 3 4 2 4 4 2 2 4 3 3 3 3 3 2
## [7549] 2 2 4 3 4 4 2 3 3 4 4 4 3 4 4 4 4 4 4 3 4 4 4 4 4 4 3 4 4 4 4 4 4 3 4 4 4 3 3
## [7586] 4 4 4 4 3 3 4 4 4 4 3 4 4 4 4 4 3 2 3 3 4 2 2 3 4 4 3 4 4 4 4 4 4 2 4 3 4 3 3 4
## [7623] 4 4 3 3 3 4 3 3 3 3 4 4 4 4 4 4 3 4 3 2 2 2 3 4 2 4 3 3 4 4 4 4 4 4 4 4 4 2
## [7660] 3 2 2 3 3 4 4 1 4 3 4 4 4 4 4 3 4 4 4 4 4 4 3 4 3 4 1 3 4 3 4 4 4 4 3 4 3 3
## [7697] 2 4 4 4 3 4 4 4 4 4 4 4 4 4 4 2 3 4 4 4 3 4 3 2 4 4 4 4 4 4 4 4 4 4 4 4 4 4
## [7734] 4 4 4 4 3 3 4 4 4 4 4 4 4 4 4 4 3 4 4 4 4 4 4 4 4 4 3 3 4 4 4 4 4 4 3 4 3 3 4 3 4
## [7771] 4 2 2 4 4 4 4 4 4 4 4 4 2 4 3 4 4 2 4 2 3 2 2 4 3 4 4 3 4 4 4 4 4 4 4 4 4 4 2
## [7808] 2 4 4 4 4 4 3 4 4 4 4 3 1 3 3 3 4 4 3 4 4 4 4 4 4 4 4 3 4 4 4 4 4 4 4 3 4 4 3
## [7845] 4 3 3 2 4 4 4 4 2 4 4 4 4 3 4 4 4 2 4 4 4 4 4 2 4 4 4 4 4 3 4 4 3 4 4 2 2 3
## [7882] 3 3 4 3 4 3 4 3 3 4 3 4 4 3 4 3 3 2 3 4 4 3 4 4 4 2 4 4 4 3 4 3 4 4 2 4 1 4 4 4
## [7919] 4 4 4 4 3 2 4 3 4 4 4 3 4 4 3 4 4 4 4 4 4 3 4 3 4 3 2 3 4 2 2 3 3 2 4 4 3 4 4
## [7956] 4 3 4 3 4 4 3 2 4 4 3 4 2 4 4 2 4 4 2 4 4 4 4 4 4 4 3 3 4 3 3 4 3 4 2 3 3 2 4 4 4 2
## [7993] 4 2 2 2 3 3 3 4 4 3 3 4 4 4 4 3 4 3 3 3 3 4 4 3 3 4 3 4 3 4 2 3 3 2 4 4 4 3
## [8030] 3 2 3 4 2 2 4 3 4 4 3 3 4 4 4 3 3 2 4 3 4 2 3 4 3 3 4 4 3 4 4 4 4 3 3 4 3 4 4 4 4 4
## [8067] 4 4 3 3 2 4 4 2 2 2 4 4 4 3 3 4 3 2 3 3 4 4 4 4 3 3 4 3 4 4 4 4 3 3 4 3 4 3 4 3
```

```

## [8104] 3 3 4 3 4 4 3 4 3 2 4 3 4 4 3 4 4 4 3 4 2 4 4 2 4 4 4 2 4 3 2 3 3 3
## [8141] 4 4 4 4 4 3 4 4 3 4 4 4 4 1 4 4 4 3 4 3 4 3 4 3 3 4 4 3 3 4 3 4 3 4 3
## [8178] 3 4 2 3 2 2 2 4 4 4 3 4 4 3 4 2 4 4 4 3 4 2 4 3 4 3 4 1 4 2 4 2 4 4 2 4 2
## [8215] 4 2 2 4 2 4 2 4 4 3 4 3 4 4 4 4 4 3 3 4 4 2 3 4 4 4 4 4 3 4 4 2 4 2 2 2
## [8252] 2 4 4 4 3 4 3 3 3 4 4 4 2 4 3 4 2 4 3 4 3 4 4 3 3 2 2 4 4 4 4 4 3 2 3 3
## [8289] 3 3 4 4 2 4 4 3 4 4 4 4 4 3 4 4 3 3 4 3 4 4 4 3 4 4 3 4 4 3 4 4 2 4 3 3
## [8326] 4 2 2 4 4 4 4 3 4 4 3 4 4 3 3 3 4 3 3 4 3 4 3 4 2 2 2 4 4 3 4 3 4 3 4 2
## [8363] 3 4 4 4 4 4 3 4 4 2 4 4 3 3 4 4 4 3 4 4 4 4 4 4 3 4 3 4 4 3 4 4 3 2 4
## [8400] 4 3 4 4 4 4 3 4 3 3 4 2 3 4 3 4 4 3 4 4 4 4 4 3 4 3 2 4 4 3 3 4 3 4 4 2 2
## [8437] 3 4 4 4 4 4 2 2 2 4 4 4 4 3 4 3 4 4 2 4 4 2 2 2 3 4 4 4 4 4 4 3 4 4 4 4
## [8474] 4 2 4 4 3 4 4 4 4 4 3 3 4 4 4 1 3 3 4 3 4 2 4 4 4 4 3 4 4 4 4 4 2 3 4 3 4
## [8511] 4 4 2 4 4 3 4 2 4 4 4 3 4 3 4 3 2 4 4 4 4 3 4 4 4 2 3 4 4 3 4 3 4 4 4 4
## [8548] 4 4 4 4 4 4 3 3 3 3 4 3 4 4 2 3 2 3 4 4 3 3 3 4 4 2 4 4 4 4 4 4 2 4 4 3
## [8585] 4 2 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 3 4 4 3 4 4 4 4 3 4 4 4 4 4 3 4 4 2
## [8622] 4 4 2 3 4 4 3 4 2 4 2 2 4 2 3 3 3 3 4 2 2 3 4 2 4 4 4 4 3 2 3 4 4 4 4 4 4
## [8659] 2 3 3 2 3 3 3 3 3 4 4 4 4 4 4 4 4 3 2 3 2 3 3 3 4 4 3 3 4 3 4 4 4 4 3 3 4
## [8696] 4 4 4 4 3 3 4 3 4 3 4 4 4 4 4 4 4 4 3 3 2 2 2 3 4 4 3 2 3 3 4 4 4 4 4 3 4
## [8733] 4 4 3 4 4 2 4 4 3 4 4 4 4 4 4 4 4 3 4 4 4 3 2 2 4 3 4 4 4 4 3 4 4 4 2 2 2 4
## [8770] 4 4 3 4 3 4 4 3 3 4 4 4 3 4 4 4 3 4 4 4 4 3 3 4 3 4 4 4 3 3 4 4 4 4 4 4 4
## [8807] 4 2 2 3 4 4 3 4 4 3 4 4 3 4 4 2 4 4 4 4 4 3 3 4 4 3 4 4 4 4 4 4 3 4 4 4 4 4
## [8844] 3 2 4 3 4 4 3 2 4 4 3 4 4 3 4 1 4 4 4 4 4 3 4 3 4 4 3 3 4 2 2 2 4 2 4 4 3
## [8881] 4 3 4 4 2 2 2 4 3 4 3 4 3 4 4 4 4 4 4 2 4 4 4 4 4 3 3 3 4 4 3 2 4 3 2 2
## [8918] 4 2 4 4 4 4 2 3 2 2 3 4 4 4 3 4 4 4 3 3 4 3 4 4 2 2 4 4 4 4 3 4 2 4 4 3 3
## [8955] 4 4 4 2 3 2 3 4 4 4 3 2 4 3 4 3 4 4 4 4 3 3 4 4 2 4 2 2 2 2 4 2 4 4 4 1
## [8992] 3 4 2 3 3 4 4 3 3 2 4 4 3 3 3 4 4 4 4 4 4 3 3 3 4 4 4 4 4 3 2 4 4 4 4 4
## [9029] 4 4 4 4 4 4 2 4 4 4 4 1 2 4 4 4 4 3 3 3 4 4 4 4 4 4 4 4 3 3 4 3 4 4 3 3 2
## [9066] 4 4 4 4 4 4 4 4 3 3 2 4 4 4 4 3 4 4 4 3 4 2 3 4 3 3 4 2 4 3 4 3 3 3 4 4 4
## [9103] 4 2 2 4 2 4 3 4 3 4 4 4 3 2 4 4 4 3 3 3 3 4 2 2 2 4 4 2 2 4 4 4 3 4 4 4 2 3
## [9140] 3 4 4 4 3 3 2 4 3 3 4 3 4 3 4 3 4 3 3 2 2 3 4 4 2 3 4 4 3 4 4 4 4 4 4 4
## [9177] 2 4 4 4 4 4 4 4 4 4 3 4 4 3 2 4 2 4 3 4 3 3 4 4 4 4 4 3 3 4 3 4 4 4 4 4 4
## [9214] 4 3 3 2 4 3 2 4 3 4 2 2 2 4 3 4 4 3 2 3 4 3 3 4 4 4 4 3 2 4 3 4 4 3 4 4 4 3
## [9251] 3 4 3 4 4 4 4 4 4 4 4 4 4 4 3 2 3 4 4 4 4 3 1 4 3 3 3 3 4 4 2 4 4 3 4 4 4 4 2
## [9288] 4 3 4 4 4 2 3 4 2 4 4 4 2 4 4 4 4 2 3 4 4 4 3 3 4 4 3 4 2 4 4 4 4 4 2 2 3
## [9325] 4 4 4 4 4 4 4 4 4 4 4 3 4 3 4 3 4 3 2 2 4 4 3 3 4 4 4 4 4 4 4 4 4 3 2 4 3
## [9362] 3 3 4 4 4 4 3 4 4 4 4 3 3 3 4 4 4 4 4 4 4 4 4 3 4 3 4 4 4 4 4 3 4 3 3 4 4 3 4
## [9399] 3 3 4 4 4 3 3 4 4 3 4 2 3 4 3 2 4 4 4 4 4 4 4 2 2 3 3 4 4 3 3 3 3 3 4 4 4 3 4
## [9436] 4 2 4 2 4 4 3 4 4 4 4 3 3 4 4 3 4 3 4 4 2 4 2 4 4 4 3 4 4 3 4 3 4 4 4 4 4 4
## [9473] 4 3 4 3 2 2 3 2 2 4 4 3 3 3 4 4 4 4 3 3 4 3 3 4 3 4 4 4 3 4 3 4 3 4 4 3 4 3 4
## [9510] 3 3 2 4 3 3 3 2 4 4 4 4 4 4 4 3 3 4 3 4 4 4 4 3 3 4 4 2 4 3 4 3 4 2 4 4 4 4
## [9547] 4 4 2 4 4 3 4 3 3 4 3 2 3 3 4 4 2 4 4 3 2 4 4 2 4 4 3 4 4 2 3 4 4 3 3 3 3
## [9584] 4 4 4 3 4 4 4 2 2 2 4 4 4 4 4 4 4 3 3 3 2 4 4 4 3 4 4 3 4 4 3 4 4 4 4 4 4 3
## [9621] 3 4 4 2 4 4 4 4 3 2 4 3 3 4 3 3 4 4 4 4 3 3 3 3 4 3 4 2 4 4 3 3 3 4 3 4 2 4 4
## [9658] 4 4 3 3 4 4 4 4 4 4 3 4 4 3 3 4 2 4 4 3 4 4 4 4 3 4 3 3 4 4 3 2 3 4 4 4 3 3
## [9695] 4 4 3 4 4 4 4 4 4 4 3 4 4 4 4 4 4 2 3 3 4 4 4 4 3 3 4 4 4 4 3 4 4 4 3 3 4 4 3
## [9732] 3 3 4 4 4 4 4 4 3 3 4 3 3 4 4 4 4 2 4 4 4 4 4 4 4 3 4 3 4 3 4 3 4 4 4 3 4 4 3
## [9769] 3 3 4 4 3 4 2 2 2 3 2 2 4 4 3 4 4 4 4 3 3 3 4 4 4 4 3 4 4 2 2 4 4 2 4 2 3 4 4 4
## [9806] 4 2 4 2 3 4 4 4 4 3 3 3 4 3 4 4 3 2 4 4 4 4 4 4 4 4 2 4 3 2 2 3 4 4 4 3 4
## [9843] 4 4 2 4 3 3 3 4 4 4 4 4 3 4 3 3 3 4 4 4 4 4 4 4 4 3 4 4 2 2 4 4 4 4 4 4 2 2
## [9880] 3 4 4 3 4 4 4 4 4 4 3 3 4 3 3 4 4 4 4 4 4 4 4 4 2 4 3 4 4 3 4 4 4 3 4 4 4 4 3
## [9917] 4 4 4 3 2 2 4 4 4 4 3 4 4 4 4 3 4 4 4 4 4 4 3 4 3 3 3 3 4 4 4 4 3 3 3 4 4 4 4
## [9954] 4 4 4 3 4 4 4 4 4 4 4 4 4 3 3 3 4 3 4 3 3 3 4 4 3 4 3 3 4 3 3 4 4 4 4 3 4 4 4
## [9991] 4 4 4 4

## Within cluster sum of squares by cluster:

```

```

## [1] 3315.203 4284.643 7571.058 3614.724
##   (between_SS / total_SS =  53.0 %)
##
## Available components:
##
## [1] "cluster"      "centers"       "totss"        "withinss"      "tot.withinss"
## [6] "betweenss"     "size"          "iter"          "ifault"

```

The within cluster sum of squares by cluster value is now 53.0%. This represents the compactness of the clustering, or how similar observations are to other observations within the same group.

We can try re-running the k-means model with the number of groups, $k = 7$

```

set.seed(123)
clustering_results_7 <- kmeans(scale(data1[, (15:18)]), centers = 7, nstart = 25)
clustering_results_7

```

```

## K-means clustering with 7 clusters of sizes 3317, 200, 12, 914, 2574, 9, 2968
##
## Cluster means:
##           Sales    Quantity   Discount     Profit
## 1 -0.17099202 -0.55401551 -0.7476979  0.01999970
## 2  3.70037129  1.13047292 -0.4332859  2.62598142
## 3  8.36049519  1.14320638  2.0689103 -11.91366497
## 4 -0.21824012  0.05129912  2.6451425 -0.46197164
## 5  0.04223395  1.22666437 -0.2946436  0.03793345
## 6 16.87915324  0.84359568 -0.6489669  19.79865024
## 7 -0.11265862 -0.54381990  0.2993725 -0.10180518
##
## Clustering vector:
## [1] 1 1 1 4 7 5 1 5 7 5 5 7 7 7 4 4 5 1 1 7 7 5 5 7 1 7 1 3 4 7 7 5 4 7 7 5 4
## [38] 5 7 7 7 7 1 7 1 1 1 1 5 5 1 1 5 5 5 5 1 1 7 1 1 5 1 1 7 5 7 1 7 1 5 7 1 7 1 5 7
## [75] 7 4 4 7 4 1 1 5 1 5 7 1 1 1 5 1 7 1 1 1 4 5 7 5 5 7 4 1 5 7 4 5 7 7 5 1
## [112] 1 5 5 7 7 5 1 4 1 1 5 5 5 7 4 4 5 7 1 7 7 5 1 1 1 7 5 5 7 1 1 1 5 7 5
## [149] 1 2 5 5 7 5 1 5 1 7 1 5 1 7 5 7 7 3 7 5 5 4 7 1 1 5 4 7 4 5 5 1 7 5 1 5 1
## [186] 1 5 7 7 7 5 5 7 5 5 7 7 1 4 7 7 4 7 4 5 7 1 1 1 5 1 7 7 4 7 4 7 5 7 7 5 4
## [223] 7 4 7 1 5 1 7 5 4 7 4 4 7 7 1 5 4 4 7 4 4 7 2 1 5 2 1 7 1 2 5 1 4 7 5 5 1
## [260] 1 5 4 7 5 7 1 7 5 7 7 1 5 1 7 1 1 7 7 5 7 4 5 5 5 5 4 7 5 5 5 7 4 5 7 7
## [297] 4 7 5 5 1 1 1 4 1 1 1 5 1 7 7 7 7 1 7 7 7 1 2 5 1 1 1 5 1 7 7 7 7 5 4 7 4
## [334] 4 7 7 7 7 7 1 7 4 5 5 7 7 1 1 1 1 1 7 1 2 5 1 1 5 5 5 1 1 1 5 1 1 1 5 1
## [371] 5 7 4 5 7 5 7 5 4 7 4 5 7 1 7 7 4 7 7 7 1 7 2 4 7 1 1 1 7 7 7 7 5 1 5 7
## [408] 5 5 1 5 1 5 7 5 5 1 1 1 5 1 1 1 5 1 5 4 5 5 7 5 4 5 7 5 5 1 1 1 1 1
## [445] 1 5 1 1 1 5 1 1 1 7 7 1 7 7 7 1 5 7 5 4 4 7 7 4 7 4 1 1 7 7 5 5 5 1 5 5
## [482] 1 1 1 1 5 5 2 7 7 7 5 1 1 7 1 5 1 7 5 4 4 7 4 7 5 1 1 1 6 1 1 7 1 1 2 1 1
## [519] 1 7 7 4 1 7 1 5 7 7 7 7 5 7 1 5 5 7 4 7 1 1 5 5 5 7 7 4 5 1 5 4 5 7 7 7 1
## [556] 7 1 7 5 1 5 5 7 1 1 7 5 5 1 7 5 1 5 1 1 5 1 1 7 4 1 7 7 7 4 7 1 1 7 7 4 7
## [593] 7 7 1 7 5 5 7 7 7 7 7 4 4 7 4 5 7 7 7 5 5 4 1 1 5 1 1 4 1 5 1 5 5 1 5
## [630] 1 7 5 7 7 7 1 4 7 5 7 1 1 5 1 1 1 5 7 1 7 1 5 1 5 7 4 7 7 4 5 5 5 4 5 1 7
## [667] 7 7 5 4 7 5 5 1 5 1 4 7 4 4 7 1 1 3 7 1 1 1 7 1 1 1 5 5 7 5 7 7 4 4 7 1
## [704] 7 7 5 7 1 5 5 1 7 5 1 1 5 1 7 7 5 1 1 1 5 5 7 7 5 7 1 1 1 1 5 7 1 5 1 7
## [741] 7 4 5 7 7 7 5 1 7 1 5 5 1 7 1 4 1 1 7 1 5 1 7 1 1 5 1 5 7 1 1 1 5 1 1 5 5
## [778] 7 1 1 1 4 7 7 7 1 7 7 1 1 5 1 5 1 1 5 5 1 5 5 1 7 5 1 1 7 2 1 1 1 5 1 1 1
## [815] 2 5 7 5 4 5 4 7 1 1 1 1 5 1 1 1 1 1 7 7 7 4 7 7 1 1 7 7 7 1 1 5 1 1 7 5 1
## [852] 5 5 1 1 1 5 1 1 1 1 5 7 7 1 1 7 4 4 7 1 1 5 5 1 1 4 7 5 1 1 5 1 7 5 5

```

```

## [889] 5 5 7 1 1 5 5 7 7 5 5 4 5 4 7 1 1 1 7 5 7 1 2 1 5 1 5 7 7 4 7 7 1 1 1 5 7
## [926] 7 5 1 1 7 7 7 7 5 4 7 7 1 1 1 1 7 7 5 1 5 4 5 5 5 7 5 7 7 7 5 7 4 7 1 7 1
## [963] 1 7 1 1 1 1 1 1 5 7 4 7 5 7 4 2 4 1 5 7 7 5 7 7 7 4 5 1 5 7 1 5 2 1 1 1 5
## [1000] 5 1 2 5 1 1 7 7 4 5 7 1 1 1 2 1 5 1 1 5 5 1 1 2 7 7 7 7 7 5 1 1 5 5 7 4 4
## [1037] 1 7 5 7 1 1 7 5 4 5 5 7 5 1 7 7 5 1 1 7 7 7 7 7 7 5 7 7 7 4 7 7 4 5 7 7
## [1074] 4 4 7 5 1 1 7 1 5 5 1 1 2 1 7 1 7 7 5 7 5 1 5 4 5 1 1 1 1 4 4 1 5 4 4 5 7
## [1111] 5 7 4 7 7 1 1 5 5 1 1 5 5 1 1 1 1 1 5 1 1 5 1 7 7 7 5 1 1 5 1 1 1 1 7 1 1
## [1148] 1 1 1 1 5 1 5 1 2 1 1 7 1 1 1 1 7 5 1 5 1 1 1 1 1 5 4 5 1 1 1 1 7 5 1 7 5
## [1185] 5 5 1 1 1 5 7 1 1 7 7 7 7 5 7 4 7 4 7 7 7 5 1 7 5 5 7 7 7 4 4 7 7 7 7 5 5
## [1222] 5 1 1 7 1 1 1 1 1 1 1 1 7 7 4 7 7 1 1 1 1 7 1 5 5 2 7 5 5 2 1 1 7 7 7 7 4
## [1259] 7 5 7 4 7 1 5 1 1 5 7 1 7 4 4 7 7 7 7 7 1 1 1 7 4 7 4 5 1 1 7 1 7 5 1 5 1
## [1296] 5 7 7 5 1 5 1 1 1 1 5 5 7 1 1 7 1 1 7 5 5 5 7 7 4 4 7 5 7 1 4 5 1 1 1 7 7
## [1333] 4 4 5 1 1 1 1 5 4 4 4 1 5 1 5 5 5 5 7 1 5 1 7 7 5 1 4 7 5 7 5 4 5 7 7 5
## [1370] 4 7 1 1 7 5 1 5 5 5 5 7 4 1 5 5 1 1 1 1 1 1 1 5 7 4 4 7 1 1 1 1 1 7 5 7 7
## [1407] 7 1 1 4 7 1 1 5 1 1 5 7 4 7 4 7 4 5 5 7 7 5 5 5 5 5 1 2 5 5 7 7 5 5 1 7 1
## [1444] 7 2 7 5 7 4 7 4 5 1 5 2 1 1 5 7 1 7 7 7 5 1 7 7 7 1 1 1 7 1 4 5 7 5 5 7 1
## [1481] 5 1 7 7 1 1 5 1 1 7 1 1 5 1 1 5 7 5 5 7 5 7 7 1 1 5 7 1 1 1 5 5 7 4 4 7 1
## [1518] 7 7 5 5 5 5 5 5 5 7 7 5 1 7 7 7 5 7 1 1 1 1 5 5 5 7 1 5 7 4 7 7 4 1 5 5
## [1555] 1 1 7 5 1 5 5 1 4 7 5 7 7 4 1 1 5 1 1 7 7 5 7 1 1 1 5 1 7 1 1 5 1 1 1 5
## [1592] 5 7 1 7 7 5 7 7 7 4 1 7 5 5 1 5 7 1 7 7 7 7 5 7 1 1 1 1 1 5 7 1 5 2 5 5
## [1629] 5 4 7 7 7 7 7 7 5 7 1 1 1 1 7 1 2 1 7 7 4 7 1 1 5 1 1 7 1 5 7 7 7 4 5 7
## [1666] 7 7 7 7 7 7 7 1 7 1 4 4 1 1 7 7 7 4 1 1 7 4 4 7 7 5 5 5 5 7 1 1 1 1 5 5 7 2
## [1703] 1 5 4 4 5 7 1 7 5 2 5 4 7 7 5 5 1 5 7 7 7 7 7 7 7 1 1 5 7 7 7 5 7 1 1
## [1740] 5 5 7 7 7 5 1 4 7 7 7 7 1 5 1 1 5 5 4 7 7 7 5 7 1 1 5 1 4 4 7 4 7 5 1 5 1
## [1777] 1 5 5 1 1 1 1 5 5 1 1 1 5 1 2 1 7 1 5 1 5 5 7 7 1 4 3 7 2 5 5 5 4 7 5 5
## [1814] 7 7 1 1 1 4 4 4 7 7 7 4 7 7 7 5 7 2 1 7 7 5 5 5 7 7 7 7 1 7 4 1 5 1 7 7 1
## [1851] 5 5 5 1 5 1 5 4 1 1 7 1 7 4 7 7 5 5 7 1 5 7 7 1 7 7 7 7 7 4 1 5 1 5 7 1 7
## [1888] 4 7 7 1 1 7 7 1 7 2 7 4 7 5 1 7 7 7 5 4 4 1 1 1 5 7 5 7 7 7 5 1 5 1 7 5 7
## [1925] 1 1 7 7 1 1 7 7 1 5 5 5 1 4 5 4 7 7 7 4 7 7 1 1 7 1 5 1 5 5 1 1 1 1 1 5
## [1962] 5 1 5 1 1 1 5 1 1 7 5 4 7 7 1 2 5 7 7 7 1 1 5 1 1 7 1 5 1 1 4 5 7 7 4 4 1
## [1999] 1 5 5 5 1 5 4 5 5 7 7 7 7 7 7 1 1 1 7 1 1 5 2 1 1 1 5 1 7 7 1 5 1 1 7 7 1
## [2036] 1 1 1 7 4 4 5 5 5 7 1 1 1 5 4 4 5 5 5 1 5 2 1 1 1 5 7 5 7 1 7 5 1 1 7 4 5
## [2073] 7 1 5 1 5 5 5 5 7 7 5 7 1 5 5 1 1 1 1 7 5 1 5 5 1 1 7 5 7 7 7 1 1 7 4 1 7
## [2110] 1 5 7 1 7 1 7 7 5 1 5 4 7 1 5 5 5 7 7 7 1 5 5 1 4 5 7 4 4 1 5 1 7 5 7 5 7
## [2147] 4 5 1 5 7 5 1 7 4 5 1 1 7 4 7 1 1 7 7 4 7 4 7 7 7 7 7 4 7 1 5 5 5 5 1 1 2
## [2184] 5 5 7 1 5 1 2 1 7 1 1 1 5 7 1 7 7 5 5 7 1 7 7 4 7 1 5 5 5 1 7 5 1 7 7 7 1
## [2221] 7 7 5 7 7 4 7 7 7 7 1 5 5 7 5 1 5 5 4 7 5 7 1 1 1 1 7 1 5 5 5 1 1 1 1 1 1
## [2258] 5 1 1 1 1 1 1 1 1 7 1 1 7 5 5 5 7 7 7 5 1 1 7 5 1 5 1 5 7 1 5 5 1 1 5 1 1
## [2295] 1 1 5 1 2 1 1 1 1 1 1 7 7 5 7 7 5 7 7 1 1 5 1 5 1 1 5 1 1 5 5 7 7 1 7 7 1
## [2332] 5 5 1 5 1 5 1 1 7 5 5 1 7 5 5 4 5 5 7 1 1 5 5 5 1 7 5 4 7 5 1 4 1 1 1 7 1
## [2369] 5 7 1 4 7 5 7 1 4 7 5 7 7 5 1 5 7 7 7 1 7 1 5 5 1 1 1 1 5 1 5 5 5 5 5 5 5
## [2406] 5 5 5 7 5 7 5 1 7 5 1 5 7 2 1 1 7 7 5 4 7 5 4 4 4 5 1 5 7 1 1 5 4 7 7 5 7
## [2443] 1 5 5 1 1 7 1 5 5 5 7 1 5 5 1 5 1 1 1 7 5 7 1 1 1 1 5 1 5 7 7 7 5 5 1 7 1
## [2480] 7 7 5 7 5 5 1 1 1 1 1 5 2 7 5 5 1 1 7 7 7 4 4 1 1 2 5 4 7 5 1 5 4 7 7 7
## [2517] 5 1 5 5 1 1 1 1 1 5 7 4 4 7 7 5 1 5 1 5 1 7 7 7 5 7 5 7 1 1 1 7 7 2 5 7 5 1
## [2554] 1 5 1 4 7 5 5 1 7 7 1 1 1 5 2 5 7 7 7 5 5 1 5 1 1 1 5 1 1 5 1 1 7 5 1 1 7
## [2591] 4 4 1 5 1 1 1 1 1 1 5 1 1 7 1 4 1 7 7 5 7 7 4 7 7 5 1 5 1 7 1 5 1 6 2 5 1
## [2628] 7 4 4 1 1 7 7 7 4 1 1 7 1 1 1 5 7 5 1 1 1 5 1 5 7 7 1 1 7 1 5 1 7 2 1 1 7
## [2665] 7 4 7 7 7 1 1 5 1 5 1 5 7 4 4 5 5 1 1 1 5 7 4 4 5 7 4 7 7 5 7 7 4 3 7 7 7
## [2702] 7 7 1 5 1 1 1 5 2 4 7 5 7 5 1 1 1 4 7 5 5 5 7 7 7 4 5 1 1 1 1 1 1 1 1 5 1
## [2739] 1 1 1 5 1 5 7 7 7 4 7 5 7 1 1 5 2 1 1 1 1 5 1 5 4 4 7 7 2 4 5 7 5 7 1 7
## [2776] 1 5 7 5 7 1 5 1 7 7 7 1 4 5 4 1 1 1 5 7 5 7 1 1 5 5 7 1 1 5 7 1 1 5 7 5 5
## [2813] 1 1 1 5 1 1 5 5 7 7 7 7 5 7 4 7 5 5 7 5 7 5 1 1 1 5 4 1 1 5 7 7 5 1 4 5 2
## [2850] 4 7 1 1 1 1 5 7 7 1 1 2 5 7 1 5 1 1 5 7 7 7 7 1 5 1 7 5 1 1 1 5 1 4 1 7 5

```

```

## [2887] 1 1 5 5 1 1 1 5 5 1 5 7 7 7 1 5 5 5 7 1 1 5 1 1 7 5 1 7 1 1 1 7 5 7 5 1 4
## [2924] 1 1 1 1 1 4 1 7 7 7 7 7 1 5 5 5 7 1 2 1 5 1 5 1 5 5 7 5 7 7 7 1 1 4 7 1
## [2961] 1 5 7 7 5 1 4 7 7 5 4 7 7 5 7 7 5 1 5 7 7 7 7 4 1 1 5 1 7 5 5 1 1 7 5 7
## [2998] 7 7 4 7 4 5 7 1 1 5 7 1 5 4 3 4 4 7 7 5 4 5 1 1 1 5 7 5 1 7 7 5 7 1 1 7 7
## [3035] 5 5 1 5 1 7 5 1 1 2 7 5 1 5 1 1 1 1 1 2 1 5 1 7 7 5 1 1 5 5 5 5 5 7 2
## [3072] 1 5 1 7 7 4 7 5 1 7 1 1 7 4 7 4 4 5 5 5 5 1 7 7 5 4 7 7 7 5 1 4 5 7 5 1
## [3109] 1 1 5 1 7 5 5 5 5 1 5 7 7 7 4 4 7 7 7 1 1 7 5 7 1 7 4 7 7 7 7 4 1 4 5 5 5
## [3146] 7 7 5 5 1 2 3 7 5 1 7 7 1 5 7 4 1 5 1 5 7 1 5 5 5 5 4 5 7 5 4 7 1 1 1 5 1
## [3183] 1 2 1 4 7 5 1 7 7 1 4 4 5 1 1 4 7 5 7 7 5 1 1 5 5 5 5 5 7 4 5 7 7 5 1 7 1
## [3220] 5 1 1 7 7 7 5 7 7 7 1 5 4 4 7 7 7 1 7 7 5 4 5 4 5 7 1 5 1 7 7 5 7 7 7 7 7
## [3257] 7 7 4 4 7 1 5 1 7 7 7 1 1 1 1 5 5 2 1 5 5 1 1 5 2 7 7 1 4 1 1 7 5 5 1 1 1
## [3294] 1 4 2 4 7 2 4 7 5 7 4 7 7 7 4 7 5 5 5 5 1 1 7 4 5 1 5 7 5 1 1 4 4 7 5 5 7
## [3331] 7 7 7 4 1 7 1 7 5 1 5 1 7 1 5 1 5 5 4 7 1 7 7 1 7 1 1 5 4 7 1 1 1 4 7 7
## [3368] 7 1 7 5 5 1 1 5 1 1 5 4 7 5 7 1 5 2 7 7 7 4 7 7 7 7 4 5 1 1 1 1 5 1 1 4 4
## [3405] 1 7 7 4 7 7 5 7 1 1 7 1 1 5 7 1 1 5 5 7 7 1 2 7 7 5 1 1 5 7 5 4 4 4 7 5 7
## [3442] 5 1 2 7 5 1 1 7 5 1 1 7 1 7 7 7 7 4 4 7 7 5 7 7 1 1 5 5 7 1 7 7 1 1 5 5 5
## [3479] 1 5 5 1 5 1 1 1 7 7 7 5 5 7 7 1 5 1 7 5 5 1 1 5 7 7 1 7 4 7 7 7 7 5 5 7 7
## [3516] 1 5 7 7 4 7 5 1 5 5 1 1 5 7 7 5 5 7 1 5 1 1 1 1 5 5 7 7 7 5 4 5 1 1 5 7 5
## [3553] 5 4 1 5 5 7 7 1 7 4 4 7 1 1 2 4 1 2 2 7 7 1 5 7 7 1 1 5 2 1 1 1 1 1 7 1
## [3590] 1 2 7 1 4 5 7 7 5 7 7 5 7 5 1 1 1 7 7 7 7 7 7 5 7 4 7 7 5 7 5 5 1 5 1
## [3627] 1 1 2 7 5 1 5 7 5 7 7 7 5 7 5 4 7 1 1 5 1 1 4 5 5 5 1 1 5 7 7 7 7 5 1 5
## [3664] 1 5 4 7 5 5 5 7 7 7 4 7 7 1 1 1 1 1 1 5 5 5 1 4 5 5 4 7 5 7 5 5 1 1 5 1
## [3701] 1 7 5 1 1 5 7 4 4 5 7 5 1 1 1 5 5 1 1 5 5 1 1 5 4 7 7 1 1 1 1 5 1 5 1 1 1 1
## [3738] 1 1 1 1 1 5 7 1 5 1 1 1 7 5 7 1 7 5 5 7 5 1 1 1 5 1 1 4 7 7 5 5 7 5 7 1 7
## [3775] 7 7 5 5 7 1 1 7 7 7 7 4 5 7 7 5 5 4 5 1 1 7 1 7 7 5 7 7 7 7 7 5 4 4 5 7
## [3812] 7 1 5 5 7 1 1 5 7 7 1 5 5 7 5 5 2 1 1 1 1 1 7 7 7 7 7 7 5 1 7 7 1 1 5 5 1
## [3849] 1 5 5 1 4 1 5 4 5 7 5 4 7 7 7 1 4 1 5 5 7 5 1 1 7 7 4 7 7 4 7 7 7 5 7 7 7
## [3886] 4 1 1 1 1 1 2 5 5 4 5 7 7 7 7 7 5 5 7 4 7 7 7 7 4 7 1 1 4 7 7 4 7 1 1 7
## [3923] 5 5 1 1 1 5 7 7 5 7 7 5 5 7 7 7 2 1 7 5 5 1 7 7 7 7 7 5 7 7 1 1 1 1 4 7
## [3960] 7 5 5 1 1 7 7 1 4 4 5 5 1 1 5 5 5 1 1 7 7 1 1 7 2 1 7 2 5 5 1 7 7 7 5 1 5
## [3997] 5 1 5 5 5 5 7 7 7 1 1 5 7 5 7 7 1 1 1 5 5 1 5 1 7 7 5 5 7 7 5 7 4 1 1 5 5
## [4034] 1 4 5 7 5 1 5 7 7 5 5 5 4 4 1 5 1 1 7 7 5 7 5 5 7 7 7 7 5 7 5 7 5 5 4
## [4071] 4 7 1 5 7 7 1 1 1 7 5 4 4 7 4 7 7 7 5 1 1 7 7 2 1 1 5 5 6 1 4 4 5 1 5 7 4
## [4108] 1 7 1 1 1 1 7 7 5 7 1 5 5 7 5 1 7 1 7 5 5 4 1 4 7 7 5 1 5 5 5 4 7 5 1 5 5
## [4145] 1 7 1 4 7 7 7 5 7 7 4 4 7 4 7 1 1 1 1 1 5 5 1 1 7 5 7 5 1 1 1 1 5 1 5 5 4
## [4182] 1 1 1 1 5 4 4 7 1 6 5 1 7 1 5 1 7 4 1 7 5 5 5 1 1 1 1 2 7 7 1 1 1 1 1 1 5
## [4219] 2 1 5 5 1 1 4 4 1 7 1 4 4 7 5 4 7 5 4 5 1 1 1 7 1 1 1 1 1 5 5 5 1 5 5 7 7 7
## [4256] 4 4 4 7 1 5 1 1 4 5 4 5 4 5 1 5 1 5 1 7 4 1 6 1 5 1 1 1 1 7 7 7 7 4 7 1 5
## [4293] 1 7 4 1 1 2 7 5 7 5 7 5 1 5 7 7 5 7 7 1 7 1 1 7 1 1 5 7 1 4 7 4 7 7 1 5 5
## [4330] 7 5 1 5 5 7 7 7 1 1 7 1 1 5 5 1 1 1 1 4 5 7 7 7 1 4 4 7 7 4 7 1 1 1 1 1 1
## [4367] 5 7 5 5 1 5 1 5 5 5 1 5 1 1 1 7 4 7 7 5 1 7 5 7 1 1 7 5 7 4 7 7 7 5 1 5 7
## [4404] 1 7 7 1 1 1 1 5 1 1 1 1 7 5 5 7 4 5 7 1 1 5 5 1 1 2 1 1 1 5 1 1 4 5 4 7 7
## [4441] 1 5 5 5 5 5 1 5 1 1 1 5 1 5 1 7 7 5 7 7 7 4 7 7 7 5 1 7 1 7 7 7 7 7 7 7
## [4478] 1 1 5 5 1 5 1 7 1 1 4 5 4 7 7 7 5 5 5 5 4 1 1 5 5 5 5 1 1 7 1 1 7 1 1 4 7
## [4515] 1 1 5 5 1 5 1 1 7 5 5 1 5 1 7 4 1 1 7 1 1 1 7 1 1 1 5 4 1 5 7 5 4 7 4 7 1
## [4552] 7 7 1 1 1 1 4 7 7 1 7 5 7 5 7 4 7 4 1 1 1 7 5 1 5 1 7 1 5 5 7 5 1 5 5 5 1
## [4589] 1 5 1 7 7 1 7 1 5 1 5 1 7 7 7 5 1 1 7 5 5 1 7 5 4 5 7 1 1 2 7 5 7 7 7
## [4626] 7 5 1 5 1 1 5 1 5 1 1 1 1 1 2 1 4 7 7 7 1 1 1 5 5 5 1 1 5 1 7 7 7 7 5 7 5
## [4663] 7 7 4 7 1 1 4 5 5 1 1 1 1 7 7 1 5 1 7 5 5 7 4 4 5 5 4 7 7 7 5 7 1 1 1 5 1
## [4700] 7 7 7 4 4 5 1 1 5 1 1 1 4 5 4 7 7 1 1 5 5 1 4 1 1 4 7 1 7 5 7 5 1 1 1 1 1
## [4737] 1 1 1 7 1 1 1 1 1 1 1 1 5 7 7 7 7 4 7 4 4 1 7 7 7 7 1 1 1 5 7 7 7 1 5 4
## [4774] 5 7 1 1 1 5 1 7 4 5 5 5 4 7 7 5 5 4 7 7 1 1 1 4 5 7 5 7 4 4 7 5 1 5 1 7 1
## [4811] 1 7 7 7 1 7 7 7 7 1 4 5 4 4 7 7 7 1 1 7 5 1 1 1 5 1 7 4 5 7 1 4 5 1 7 7 5
## [4848] 5 7 1 5 5 5 1 1 1 7 1 5 5 2 5 7 7 1 2 5 1 1 7 5 7 4 7 4 5 7 7 4 1 1 2 1 5

```

```

## [4885] 1 5 1 5 1 5 1 1 1 1 1 1 1 1 1 7 7 4 1 1 5 1 7 1 5 4 7 7 5 7 4 4 1 7 5 7 1 1 4
## [4922] 1 1 5 1 1 7 1 1 5 1 1 7 4 7 7 5 5 7 5 5 1 1 1 1 7 1 1 5 1 1 1 1 5 1 1 4
## [4959] 5 7 1 7 1 1 4 7 1 1 1 1 7 7 7 7 1 5 5 7 5 7 4 7 1 1 5 5 7 5 4 7 7 3 5 5 5
## [4996] 7 7 4 7 5 1 5 1 1 1 5 2 1 4 5 4 4 4 5 5 5 7 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 5
## [5033] 1 1 5 1 1 5 7 1 5 1 1 5 5 1 7 5 4 7 1 7 1 5 7 5 5 5 4 1 5 1 1 1 1 1 7 5 5 1
## [5070] 1 1 5 1 1 7 5 5 7 1 7 1 7 7 7 1 4 5 7 4 7 7 7 1 4 4 7 4 4 4 7 7 7 7 5 7 7
## [5107] 1 5 5 5 5 7 4 7 7 7 1 1 5 5 5 1 1 1 5 5 2 1 1 1 5 5 7 5 7 5 7 1 1 5 1 5 5
## [5144] 7 1 1 1 7 7 1 1 5 7 1 1 1 7 7 4 7 7 5 1 7 1 1 1 1 1 2 1 1 7 4 7 7 7 1 7
## [5181] 1 5 1 1 5 5 5 1 2 5 7 1 5 5 1 1 1 5 2 7 7 5 5 1 1 1 1 1 5 1 1 1 1 1 5 5 1
## [5218] 1 5 5 1 5 5 5 1 5 5 7 1 7 5 1 5 5 7 1 1 1 1 1 7 1 4 1 7 7 7 4 1 7 5 1 1 7
## [5255] 7 7 1 5 5 7 7 4 7 7 7 1 1 1 1 1 5 1 5 4 1 1 5 5 1 1 1 1 1 1 5 4 4 1 7 5
## [5292] 5 7 7 5 7 1 5 1 1 2 2 5 7 4 5 1 1 7 5 3 7 7 1 1 5 7 7 7 4 4 4 7 1 5 5 1 1
## [5329] 7 7 1 5 1 1 1 1 7 5 5 5 1 5 7 5 1 1 7 7 1 5 5 7 7 5 5 1 5 1 1 1 1 5 5 5 1 7
## [5366] 7 7 1 1 7 1 1 7 5 7 5 1 5 7 1 5 5 5 7 7 5 5 7 1 1 7 5 1 5 5 5 5 4 5 1 1
## [5403] 7 7 5 1 1 1 7 7 1 5 1 5 4 7 7 1 5 7 7 7 7 5 1 2 5 5 1 7 7 7 4 7 2 1 7 4 7
## [5440] 7 7 7 1 5 1 7 5 5 1 5 1 1 5 5 7 7 5 1 5 5 7 1 4 7 5 4 7 7 7 1 5 5 7 4 1 1
## [5477] 5 1 1 4 7 1 1 1 7 7 4 7 7 7 7 7 1 1 7 4 7 1 1 1 5 1 7 7 7 7 1 1 5 7 7 7 7
## [5514] 1 1 1 2 5 7 7 5 7 5 1 5 1 1 1 5 7 2 5 1 1 1 1 1 1 7 4 7 4 4 4 5 1 7 1 5
## [5551] 7 5 7 5 7 1 1 1 1 5 1 1 2 1 7 7 5 7 7 7 7 4 7 7 7 5 1 1 5 5 1 5 1 1 1 1
## [5588] 5 7 1 1 1 1 4 1 1 1 1 1 7 5 7 7 5 5 7 7 5 1 1 7 1 5 1 5 5 5 7 1 5 1 1 5 1
## [5625] 1 1 2 5 7 5 1 5 1 5 5 5 1 1 5 7 5 1 1 1 5 5 5 1 5 5 1 5 1 1 4 7 7 7 4 7 7
## [5662] 7 5 7 4 7 7 1 5 1 1 7 7 7 1 5 1 4 5 1 1 1 1 5 5 7 5 5 5 7 5 5 1 5 1 7 4 4
## [5699] 7 5 7 5 7 7 7 1 7 1 1 1 2 1 7 1 5 5 1 5 1 5 5 7 7 1 7 5 5 1 1 1 1 5 1 1
## [5736] 7 1 1 7 1 1 7 1 1 5 5 7 7 7 5 4 1 5 1 1 4 2 7 4 1 7 1 1 5 5 7 5 5 1 1 1 1
## [5773] 1 7 5 7 7 1 1 7 7 1 1 1 4 5 4 5 5 4 5 7 1 1 7 2 1 1 1 7 7 7 7 7 7 7 7 7 7
## [5810] 7 1 1 7 1 7 1 7 7 5 1 1 1 5 7 7 1 1 1 1 1 1 7 5 4 4 5 7 5 5 1 7 5 1 5 5
## [5847] 7 1 1 1 1 1 1 5 7 4 4 7 1 7 1 1 1 1 5 7 4 7 5 1 1 5 1 7 5 1 5 1 7 5 5 1
## [5884] 1 2 7 1 1 1 1 7 7 7 1 1 1 5 1 1 1 5 1 1 1 5 1 1 7 1 1 7 1 1 5 5 1 1 2 5 1
## [5921] 1 1 1 7 5 5 5 1 1 4 7 7 7 7 5 7 7 7 7 5 1 7 1 5 1 1 5 1 1 1 5 7 5 7 4 5
## [5958] 5 1 7 1 5 1 1 1 5 7 1 1 7 1 1 7 1 5 1 1 5 5 1 1 1 1 5 5 1 1 5 1 5 2 5 7 5
## [5995] 7 7 7 7 7 1 1 7 4 4 1 1 1 1 1 2 1 7 5 2 1 5 1 1 5 1 1 7 1 5 7 5 5 7 1 7
## [6032] 1 5 1 4 4 1 5 1 1 7 7 7 7 5 7 1 4 4 5 5 5 7 7 4 4 5 1 5 7 1 1 5 1 1 5 5 5
## [6069] 1 7 7 7 7 7 7 1 5 1 1 1 5 4 4 5 7 7 4 1 1 4 1 1 5 1 1 1 1 7 1 2 1 2 1 1 7
## [6106] 7 4 5 7 4 5 7 5 5 7 5 2 1 7 7 5 4 7 7 5 1 7 5 7 1 1 5 5 7 1 7 4 5 7 1 5 1
## [6143] 1 5 1 1 1 7 7 5 5 5 1 7 7 7 7 1 5 1 7 7 7 4 7 7 7 7 1 1 7 1 5 1 1 2 5 1 1
## [6180] 1 5 1 1 1 5 5 5 1 5 5 7 5 1 5 4 5 7 7 5 1 1 2 7 5 5 7 5 7 7 2 5 1 1 1 1 1
## [6217] 5 1 5 1 5 5 7 5 7 4 7 7 7 7 1 7 1 5 2 5 7 7 1 1 7 5 4 1 5 1 1 1 1 1 1 4 5
## [6254] 1 5 7 4 7 1 1 7 1 4 1 5 7 5 1 5 7 5 5 5 1 1 5 5 1 7 7 7 7 7 5 5 7 7 7 5 7
## [6291] 5 1 7 1 7 2 5 1 7 1 4 7 5 7 1 5 7 7 7 5 7 1 1 1 1 1 1 5 5 7 7 1 1 5 1 1
## [6328] 1 5 1 1 1 1 5 1 5 1 1 2 7 1 1 1 1 5 7 1 1 4 7 4 1 1 5 5 1 1 5 1 5 5 7
## [6365] 5 1 1 1 5 4 7 1 1 1 7 5 7 1 1 5 5 4 7 4 7 5 4 1 1 5 1 1 1 5 1 1 7 7 5 5 5
## [6402] 1 1 1 7 7 5 5 5 7 7 7 1 5 4 7 7 7 1 1 1 1 5 5 1 2 4 7 7 7 5 4 7 5 5 1 5 7
## [6439] 7 5 5 1 1 5 1 5 4 7 7 1 1 1 7 7 1 5 1 5 1 5 5 1 1 5 1 5 5 2 5 5 1 5 5 7 4
## [6476] 4 7 7 7 5 5 7 7 7 5 1 7 4 5 5 7 7 5 7 7 7 1 5 5 5 5 5 4 1 7 7 7 5 7 1 5
## [6513] 1 5 7 7 5 5 1 7 2 1 5 7 7 4 4 2 1 7 4 5 1 5 2 2 7 7 1 5 5 1 7 1 4 7 7 5 7
## [6550] 5 1 1 5 7 7 7 4 7 5 5 1 4 1 7 1 1 4 5 7 7 7 7 1 1 5 5 7 5 5 1 1 1 1 5 1 5
## [6587] 5 1 1 1 7 1 5 7 5 4 7 4 7 1 1 5 5 7 1 5 7 5 7 5 7 1 1 5 1 7 5 1 5 5 2 1 5
## [6624] 5 7 7 2 1 5 4 7 2 1 1 4 7 5 1 5 5 5 1 5 7 5 7 4 7 4 7 5 4 7 7 7 7 7 5 7 5
## [6661] 1 1 5 5 5 5 5 1 1 5 1 7 5 1 5 1 5 1 1 1 1 5 7 4 5 1 1 5 1 5 5 1 1 7 7
## [6698] 1 1 7 7 7 4 7 1 7 5 1 7 7 5 7 4 4 7 5 5 1 1 7 5 1 1 5 1 1 7 5 5 4 7 7 7 7
## [6735] 7 1 1 5 7 7 7 7 7 5 5 5 5 7 1 1 1 1 1 1 1 1 1 5 1 5 1 7 1 1 5 7 7 7 7 4 5 5 5
## [6772] 4 7 7 5 1 7 5 1 1 1 5 7 7 7 5 7 1 5 7 5 1 1 7 7 4 4 4 4 4 7 4 7 7 7 1 1 7 7
## [6809] 4 4 4 7 4 5 4 1 5 2 4 4 7 4 5 5 1 1 6 1 5 1 1 1 7 7 1 5 1 1 2 5 4 1 1 5 1
## [6846] 7 1 7 7 1 2 5 1 1 1 1 1 1 5 5 5 7 7 7 5 1 7 2 1 5 7 1 4 7 1 2 4 7 2 5 2

```

```

## [6883] 1 5 2 7 7 1 7 5 5 1 1 1 1 1 5 4 5 4 5 2 7 4 5 1 1 7 1 4 1 1 1 7 7 4 7 5 1
## [6920] 4 5 1 5 1 5 5 1 1 1 5 1 7 7 7 7 7 4 7 1 7 5 1 1 5 1 5 1 7 7 5 7 7 7 5 1
## [6957] 5 1 1 1 1 1 5 1 5 1 1 5 1 5 7 7 7 4 7 7 7 1 5 5 1 1 5 1 1 1 1 5 5 4 1 5 1
## [6994] 5 5 1 4 7 1 1 1 7 5 7 7 1 4 4 7 7 5 1 1 7 1 1 1 1 5 7 7 7 5 5 5 5 7 1 5 1
## [7031] 1 1 1 7 7 7 7 1 1 1 7 1 7 7 7 7 7 1 7 7 5 1 1 1 5 7 5 5 1 1 7 7 7 7 1 5 5
## [7068] 4 5 1 7 7 7 5 4 7 5 4 5 7 7 7 4 1 5 4 7 1 1 7 1 1 1 5 5 5 7 5 5 5 7 7 4 1
## [7105] 5 7 5 1 1 1 1 1 5 7 2 7 1 7 1 7 4 5 1 1 1 1 1 5 1 7 5 1 2 1 1 1 4 7 4 7 1
## [7142] 5 5 7 4 7 7 1 1 7 1 4 7 4 5 5 1 1 1 7 7 1 5 5 7 1 1 7 5 7 1 1 5 2 7 4 7 4
## [7179] 4 1 1 7 5 5 7 7 7 4 4 7 7 4 5 5 1 1 5 1 1 1 7 5 5 4 1 1 5 7 7 7 1 5 7 1 5
## [7216] 1 1 7 1 7 4 1 7 7 5 1 5 5 5 1 1 1 1 5 1 7 1 4 5 5 5 5 1 2 5 5 7 4 5 1 1 7
## [7253] 1 1 5 1 1 5 5 5 1 5 7 7 5 7 4 7 7 5 7 1 7 4 5 5 5 7 1 5 2 4 4 5 7 1 1 1 7
## [7290] 5 7 5 4 5 1 7 5 5 4 4 5 4 5 7 7 7 1 7 1 1 5 4 4 5 7 1 7 7 4 5 5 5 7 7 1 1 5
## [7327] 7 1 2 2 4 7 4 7 7 1 7 1 1 1 1 5 1 5 4 7 7 7 4 5 5 5 5 5 5 7 7 5 7 1 7 7
## [7364] 7 5 4 7 5 4 7 1 5 1 5 1 7 7 5 7 1 5 5 1 1 1 1 5 5 1 7 5 7 5 1 5 7 1 5 5 7
## [7401] 1 5 1 5 1 7 7 5 5 7 1 7 7 7 7 5 7 5 4 5 4 7 4 7 7 7 4 1 1 1 1 1 7 4 7 5 5
## [7438] 5 5 5 7 1 7 7 7 7 4 5 5 7 7 4 5 7 7 7 1 1 5 5 7 1 5 1 5 1 5 1 1 1 4 7 2 7
## [7475] 2 1 1 1 1 1 1 1 7 5 1 5 5 2 7 1 5 5 7 7 4 7 1 1 1 1 1 7 1 7 1 4 7 1 7 5 5
## [7512] 5 5 5 1 5 1 5 1 1 2 4 4 7 5 5 5 1 5 7 7 7 7 5 1 4 1 7 4 4 7 5 5 5 5 5 4
## [7549] 4 4 1 5 7 1 4 5 5 1 1 1 5 7 7 7 1 7 7 5 1 1 5 1 1 1 1 7 1 2 1 7 1 2 5
## [7586] 1 1 1 1 5 5 7 7 7 5 1 7 7 7 5 4 5 5 1 4 4 5 7 1 5 1 1 1 1 1 1 4 1 5 7 5 5 7
## [7623] 1 1 5 5 5 7 5 5 5 5 5 1 1 1 1 1 5 1 5 4 4 4 5 7 4 7 7 5 7 7 7 1 1 1 1 1 4
## [7660] 5 4 4 5 5 1 1 2 1 5 1 1 1 1 5 7 1 7 1 1 2 1 5 1 2 5 1 2 1 5 1 1 1 5 7 5 5
## [7697] 4 1 1 1 5 7 1 1 1 1 1 1 7 4 5 7 7 7 5 7 5 4 1 1 1 7 1 7 7 7 7 1 7 7 7 7 7
## [7734] 7 1 1 7 5 5 1 1 7 1 1 1 1 7 5 1 1 1 1 1 7 7 5 5 1 1 7 7 7 7 5 1 2 5 7 5 7
## [7771] 1 4 3 7 1 1 1 1 7 1 1 4 7 5 7 1 4 7 4 5 4 4 7 5 1 7 5 1 1 1 7 1 7 7 7 7 4
## [7808] 4 7 7 1 1 7 5 7 7 1 5 2 5 5 5 1 1 5 1 7 7 7 1 7 7 7 5 7 7 7 1 1 1 1 1 5
## [7845] 7 5 2 4 7 7 7 7 4 1 1 1 7 5 1 1 4 1 1 7 7 7 4 1 1 7 7 1 7 1 1 5 7 7 4 7 5
## [7882] 5 5 7 5 7 5 5 5 1 5 1 7 5 1 5 5 4 5 7 7 5 7 7 7 4 1 7 5 1 7 4 7 2 1 7 7
## [7919] 7 1 1 1 5 4 1 5 1 1 1 5 1 7 5 1 7 1 7 2 1 5 7 5 4 5 7 4 4 5 5 4 7 7 1 1 1
## [7956] 7 5 1 5 7 7 5 4 7 7 5 1 4 7 7 7 7 4 7 4 7 1 1 7 7 1 5 5 7 5 5 1 1 1 1 4
## [7993] 7 4 4 4 5 5 5 1 1 5 5 1 1 1 5 1 1 2 5 5 5 5 1 1 5 5 7 5 1 4 5 7 4 1 1 1 1
## [8030] 5 4 5 7 4 4 7 5 1 7 5 5 1 1 1 5 5 4 7 5 4 7 5 1 4 5 7 5 5 7 1 5 7 1 1 1 1
## [8067] 7 1 5 5 4 7 7 4 4 4 1 1 1 5 5 7 5 4 5 5 7 7 7 5 5 1 5 1 1 1 5 5 1 1 2 1 2
## [8104] 5 5 7 5 7 7 5 7 7 4 7 5 1 1 5 7 1 7 5 7 7 7 4 7 7 4 1 1 1 1 4 1 5 4 5 5 5
## [8141] 7 1 1 7 7 5 1 1 5 1 1 1 6 1 1 7 5 7 5 1 5 1 5 5 7 7 5 2 7 1 5 1 5 1 7 5
## [8178] 5 1 4 5 7 4 4 7 7 7 5 1 1 5 1 4 7 7 1 7 5 7 4 7 5 1 1 2 1 7 7 4 7 7 4 7 4
## [8215] 7 4 4 7 4 7 4 7 7 5 1 5 1 1 1 1 5 5 1 7 4 2 7 7 7 7 1 5 1 1 4 7 7 4 4
## [8252] 4 7 1 1 5 1 5 5 5 1 1 1 4 1 5 1 7 7 2 7 2 7 7 5 5 4 4 7 1 7 7 7 7 5 4 5 5
## [8289] 5 5 7 7 4 7 1 5 1 7 7 1 1 2 7 7 7 5 5 1 5 7 7 1 2 7 7 5 7 7 5 1 7 4 1 5 5
## [8326] 7 4 4 1 1 1 1 5 1 1 5 1 1 5 5 5 1 5 5 7 7 5 1 5 5 1 7 4 7 4 7 1 5 7 5 7 4
## [8363] 5 7 7 7 1 1 5 7 7 4 1 1 5 5 7 7 7 5 7 7 7 1 1 7 1 5 1 5 1 1 1 5 1 7 5 4 7
## [8400] 7 5 1 1 1 1 5 1 5 5 7 7 5 7 5 7 1 5 1 1 1 1 1 5 1 2 4 7 1 5 5 1 5 7 7 4 4
## [8437] 5 7 7 7 7 7 7 4 7 7 1 1 7 5 1 5 7 7 4 7 7 4 4 4 5 1 1 7 7 1 1 7 2 7 7 7 7
## [8474] 7 4 7 7 1 1 1 1 7 1 5 5 1 1 1 6 5 5 1 5 7 4 7 1 1 1 5 1 7 7 1 7 4 5 7 5 1
## [8511] 1 7 4 7 1 5 1 4 1 1 1 5 1 5 7 5 4 7 7 7 1 1 5 1 1 1 4 5 7 7 5 7 5 1 1 1 1
## [8548] 1 1 7 7 7 7 5 5 5 5 7 5 7 7 4 2 4 5 7 7 5 5 5 7 7 7 1 1 7 1 1 1 1 4 7 7 5
## [8585] 7 4 5 5 7 7 1 7 7 7 7 1 1 7 7 1 1 7 5 7 7 5 1 1 1 7 1 5 1 1 1 7 1 7 5 1 4
## [8622] 7 7 4 5 7 1 5 7 7 1 4 4 7 4 5 5 5 5 1 4 4 5 7 4 7 7 7 5 4 5 1 1 1 1 1 1 1
## [8659] 4 5 5 4 5 5 5 5 5 7 7 1 7 7 7 7 7 5 4 7 4 5 2 5 1 1 5 5 1 5 1 7 7 7 5 5 1
## [8696] 7 7 7 7 2 5 7 5 7 5 1 1 1 7 7 1 7 7 5 5 4 4 4 5 7 5 4 5 5 1 7 7 7 7 5 1
## [8733] 1 1 5 1 1 4 7 7 5 1 7 7 1 1 1 7 7 2 1 7 7 5 4 4 1 5 1 1 2 5 1 1 7 4 4 4 7
## [8770] 7 7 7 7 5 7 7 5 5 7 7 7 5 7 1 7 5 7 7 7 5 5 1 5 5 7 1 1 5 2 7 1 1 1 1 7
## [8807] 1 4 7 5 1 1 5 7 1 5 1 1 5 7 7 4 7 7 7 7 5 5 1 1 5 1 7 1 1 1 5 1 7 1 1 1
## [8844] 5 4 7 5 7 7 5 4 7 1 5 7 7 5 7 2 7 1 1 1 1 5 1 7 7 7 5 5 1 4 4 4 4 7 4 1 1 5

```

```

## [8881] 1 5 7 7 4 4 4 1 5 1 2 1 2 7 1 1 7 1 1 1 4 7 7 7 1 1 5 5 5 1 7 5 4 7 5 4 4
## [8918] 7 4 1 1 7 7 4 7 4 4 5 7 7 7 5 1 1 7 5 5 7 5 7 4 4 7 1 1 1 5 1 5 7 7 5 5
## [8955] 7 1 7 4 5 4 5 7 1 1 5 4 7 5 7 5 7 1 7 5 5 7 7 4 1 4 4 4 4 4 4 7 4 7 7 1 2
## [8992] 5 7 4 5 5 7 7 5 7 4 1 1 5 5 5 1 1 1 7 1 7 5 5 5 7 7 1 7 1 5 4 7 1 1 1 1 1
## [9029] 1 7 1 1 1 2 4 7 1 1 7 6 7 7 1 1 7 5 5 5 1 1 1 7 7 1 1 2 5 7 5 7 1 5 5 4
## [9066] 7 1 1 7 1 7 1 1 5 5 4 7 1 1 1 5 1 1 1 5 7 4 7 7 5 5 7 4 1 5 1 5 5 5 7 7 7
## [9103] 7 4 4 7 7 7 5 7 5 1 1 1 5 7 7 7 5 5 5 1 4 4 4 1 7 4 4 4 7 1 1 5 7 7 7 4 5
## [9140] 5 1 1 7 5 5 4 1 5 2 1 5 7 7 5 7 5 1 5 5 4 4 5 1 7 4 2 1 1 5 1 1 1 1 7 1
## [9177] 4 1 1 7 7 7 1 1 1 7 2 7 7 5 4 7 4 7 5 7 1 5 5 7 1 1 7 5 5 1 5 7 1 1 1 1 1
## [9214] 1 5 5 4 1 5 4 1 2 1 4 4 4 4 7 5 1 7 5 4 5 1 7 5 7 7 7 5 4 7 5 7 1 5 1 1 7 5
## [9251] 5 7 5 1 1 1 1 1 1 1 7 1 5 4 5 1 1 1 5 2 1 1 2 5 5 7 7 4 7 1 5 7 7 1 7 4
## [9288] 7 5 1 1 1 4 5 7 4 7 7 7 4 7 7 7 7 4 5 1 1 1 5 5 7 7 5 1 7 7 7 1 1 1 4 4 5
## [9325] 7 7 1 7 1 1 1 7 1 1 7 7 5 7 5 1 5 7 5 7 4 7 7 5 5 1 7 7 7 7 1 1 1 5 7 1 5
## [9362] 5 5 1 1 1 7 5 1 1 7 7 5 5 5 1 1 1 7 7 7 7 5 1 5 1 1 1 1 5 7 5 5 1 7 5 1
## [9399] 2 5 1 7 7 2 5 7 7 5 1 4 5 7 2 4 7 1 1 7 7 4 4 5 5 7 1 2 5 5 5 5 7 7 7 5 7
## [9436] 7 4 1 7 7 1 5 7 7 7 1 5 5 1 7 5 1 5 1 7 7 7 4 1 7 1 5 1 7 5 1 5 7 1 1 1 7
## [9473] 1 5 7 5 4 4 5 4 4 7 7 5 5 5 7 1 7 5 2 1 5 2 1 5 1 1 1 5 1 1 5 1 1 5 1 1 5 1
## [9510] 5 5 4 7 2 5 5 7 7 7 1 1 7 7 7 5 5 7 5 7 1 7 5 5 1 7 4 7 5 1 5 7 4 7 7 1
## [9547] 1 7 4 7 7 5 1 5 5 7 5 4 5 5 7 7 4 7 1 5 4 7 1 4 7 7 7 5 1 7 7 7 1 1 5 5 5
## [9584] 7 1 7 5 7 7 7 4 4 4 1 1 1 7 7 1 7 5 5 5 7 1 7 7 5 1 1 5 1 7 1 5 2 1 7 7 5
## [9621] 5 7 7 4 7 7 7 1 5 4 7 5 5 1 5 2 1 1 1 3 5 5 5 7 5 7 4 1 1 2 5 1 5 7 4 1 7
## [9658] 1 1 5 2 1 7 1 1 1 1 5 1 1 5 5 7 4 7 1 5 1 1 1 5 1 5 5 1 1 5 4 5 1 1 7 5 5
## [9695] 1 7 5 7 1 1 7 1 1 5 1 7 7 1 1 1 7 4 5 5 7 7 7 5 5 7 1 1 1 5 7 1 5 5 1 7 5
## [9732] 5 5 1 1 7 1 1 5 5 1 2 5 1 7 7 7 4 7 1 7 7 7 1 1 1 1 5 1 5 1 1 1 5 1 1 5 1
## [9769] 5 5 7 7 5 1 3 4 4 5 4 4 1 7 5 1 1 1 5 5 7 7 7 5 7 7 4 4 7 7 7 7 5 7 1 1
## [9806] 7 4 7 4 5 7 1 1 5 5 5 7 5 7 7 5 4 1 1 7 7 7 1 7 7 7 4 7 5 4 4 5 1 1 7 5 1
## [9843] 7 7 4 7 5 5 5 1 1 7 1 5 1 1 5 2 5 1 1 1 1 1 5 1 7 4 4 7 7 1 7 1 7 4 7
## [9880] 5 7 1 5 1 1 1 1 1 1 5 5 1 5 5 7 7 7 7 7 7 4 7 5 7 7 5 1 1 5 7 1 1 5
## [9917] 1 1 1 7 4 4 7 7 1 5 1 1 1 2 7 7 5 1 7 7 1 7 5 1 5 5 5 5 1 7 7 2 2 1 1 1 7
## [9954] 1 7 7 5 1 1 7 1 7 7 7 1 1 5 5 1 5 1 5 7 5 5 1 7 5 7 5 1 5 5 1 5 1 1 1 5 7
## [9991] 1 7 1 1
##
## Within cluster sum of squares by cluster:
## [1] 1131.9138 1960.3599 1460.2766 1749.2333 3515.6513 778.7336 1329.8103
## (between_SS / total_SS = 70.2 %)
##
## Available components:
##
## [1] "cluster"      "centers"       "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"          "iter"          "ifault"

```

Re-running k-means with k = 8:

```

set.seed(123)
clustering_results_8 <- kmeans(scale(data1[, (15:18)]), centers = 8, nstart = 25)
clustering_results_8

```

```

## K-means clustering with 8 clusters of sizes 912, 2114, 2923, 12, 9, 182, 3263, 579
##
## Cluster means:
##           Sales   Quantity   Discount      Profit
## 1 -0.21598994  0.03592803  2.6411836 -0.463442556
## 2  0.06144353  0.84777662 -0.3020867  0.051473515

```

```

## 3 -0.12740261 -0.55718427  0.2964566 -0.102300730
## 4  8.36049519  1.14320638  2.0689103 -11.913664966
## 5 16.87915324  0.84359568 -0.6489669  19.798650239
## 6  3.85740734  0.99833967 -0.4172776  2.749990730
## 7 -0.18963127 -0.56474906 -0.7485896  0.008899425
## 8  0.17956725  2.49300690 -0.2367609  0.083089163
##
## Clustering vector:
## [1] 7 7 7 1 3 2 7 2 3 2 8 2 3 3 1 1 2 7 7 3 3 2 2 3 7 3 7 4 1 3 3 2 1 3 3 2 1
## [38] 8 3 3 3 3 7 3 7 7 7 7 2 2 7 7 2 2 2 2 2 7 7 3 7 7 2 7 7 2 8 3 7 3 7 8 3
## [75] 3 1 1 3 1 7 7 2 7 2 3 7 7 7 2 7 3 7 7 7 7 1 2 3 2 2 3 1 7 2 3 1 8 3 3 2 7
## [112] 7 2 8 3 3 2 7 1 7 7 2 8 2 3 1 1 2 3 7 3 3 8 7 7 7 7 3 2 8 3 7 7 7 7 2 3 8
## [149] 7 6 2 2 3 2 7 2 7 3 7 2 7 3 2 3 3 4 3 8 2 1 3 7 7 2 1 3 1 2 2 7 3 2 7 2 7
## [186] 7 8 3 3 2 2 8 3 2 2 3 3 7 1 3 3 1 3 1 2 3 7 7 7 8 7 3 3 1 3 1 3 2 3 3 2 1
## [223] 3 1 3 7 2 7 3 8 1 3 1 1 3 3 7 2 1 1 3 1 1 3 6 7 2 6 7 3 7 6 2 7 1 3 2 2 7
## [260] 7 2 1 3 8 3 7 3 2 3 3 2 2 7 3 7 7 3 3 2 3 1 2 2 2 2 2 1 3 2 2 8 3 1 8 3 3
## [297] 1 3 2 8 7 7 7 1 7 7 7 2 7 3 3 3 3 7 3 3 3 7 6 2 7 7 7 2 7 3 3 3 3 8 1 3 1
## [334] 1 3 3 3 3 3 7 3 1 8 8 3 3 7 7 7 7 3 7 6 2 7 7 2 2 8 7 7 7 8 7 7 7 2 2
## [371] 2 3 1 8 3 8 3 2 1 3 1 2 3 7 3 3 1 3 3 3 7 3 6 1 3 7 7 7 3 3 3 3 3 2 7 2 3
## [408] 2 2 7 8 7 8 3 2 3 8 2 7 7 7 2 7 7 2 7 8 7 2 1 2 2 3 2 1 2 3 2 8 7 7 7 7 7
## [445] 7 2 7 7 7 2 7 7 7 3 3 7 3 3 3 7 8 3 2 1 1 3 3 1 3 1 7 7 3 3 2 2 2 2 7 2 2
## [482] 7 7 7 7 2 2 6 3 3 3 2 7 7 3 7 8 7 3 8 1 1 3 1 3 2 7 7 7 5 7 7 3 7 7 6 7 7
## [519] 7 3 3 1 7 3 7 2 3 3 3 3 2 3 7 2 2 3 1 3 7 7 2 8 8 3 3 1 2 7 8 1 2 3 3 3 7
## [556] 3 7 3 2 7 2 2 3 7 7 3 8 2 7 3 2 7 2 7 7 8 7 7 3 1 7 3 3 3 1 3 7 7 3 3 1 3
## [593] 3 3 7 3 2 8 3 3 3 3 3 3 1 1 3 1 8 3 3 3 3 8 8 1 7 7 8 7 7 1 7 2 7 8 2 7 2
## [630] 7 3 2 3 3 3 7 1 3 2 3 7 7 2 7 7 7 2 3 7 3 7 8 7 8 3 1 3 3 1 2 8 2 1 2 7 3
## [667] 3 3 2 1 3 8 2 7 2 7 1 3 1 1 3 7 7 4 3 7 7 7 3 7 7 7 7 2 8 3 2 3 3 1 1 3 7
## [704] 3 3 2 3 7 2 8 7 3 2 7 7 2 7 3 3 2 7 7 7 2 2 3 3 8 3 7 7 7 7 7 2 3 7 2 7 3
## [741] 3 1 2 3 3 3 8 7 3 7 8 2 7 3 7 1 7 7 3 7 8 7 3 7 7 2 7 2 3 7 7 7 2 7 7 2 2
## [778] 3 7 7 7 1 3 3 3 7 3 3 7 7 8 7 2 7 7 2 2 7 2 2 7 3 8 7 7 3 6 7 7 7 2 7 7 7
## [815] 6 8 3 8 1 2 1 3 7 7 7 7 2 7 7 7 7 7 3 3 3 1 3 3 7 7 3 3 3 7 7 2 7 3 2 7
## [852] 2 2 7 7 7 8 7 7 7 7 7 2 3 3 7 7 3 1 1 3 7 7 2 2 7 7 7 1 3 2 7 7 2 7 3 8 2
## [889] 2 2 3 7 7 8 2 3 3 2 8 1 2 1 3 7 7 7 3 2 3 2 6 7 2 7 8 3 3 1 3 3 7 7 7 2 3
## [926] 3 2 7 7 3 3 3 3 8 1 3 3 7 7 7 7 3 2 2 7 2 1 2 2 8 3 2 3 3 3 2 3 1 3 7 3 7
## [963] 7 3 7 7 7 7 7 7 2 3 1 3 2 3 1 6 1 7 2 3 3 2 3 3 3 1 2 7 2 3 7 2 6 7 7 7 2
## [1000] 2 7 6 2 7 7 3 3 1 2 3 7 7 7 8 7 2 7 7 2 2 7 7 6 3 3 3 3 3 2 7 7 8 2 3 1 1
## [1037] 7 3 2 3 2 7 3 2 1 8 2 3 8 7 3 3 8 7 7 3 3 3 3 3 3 2 3 3 3 1 3 3 1 2 3 3
## [1074] 1 1 3 2 7 7 3 7 8 8 7 7 6 7 3 7 3 3 8 3 2 7 2 1 2 7 7 7 7 1 1 7 2 1 1 8 3
## [1111] 2 3 1 3 3 7 7 2 2 7 7 2 2 7 7 7 7 2 7 7 2 7 3 3 3 8 7 7 2 7 7 7 7 3 7 7
## [1148] 7 7 7 7 2 7 2 7 6 7 7 3 7 7 7 7 3 2 7 8 7 7 7 7 7 2 1 2 7 7 7 7 3 2 7 3 2
## [1185] 2 2 7 7 7 8 3 7 7 3 3 3 3 8 3 1 3 1 3 3 3 2 7 3 2 2 3 3 3 1 1 3 3 3 3 2 8
## [1222] 8 7 7 3 7 7 7 7 7 7 7 3 3 1 3 3 7 7 7 7 3 7 8 2 6 3 2 8 8 7 2 3 3 3 3 1
## [1259] 3 8 3 1 3 7 2 7 7 2 3 7 3 1 1 3 2 3 3 3 7 7 7 3 1 3 1 2 7 7 3 7 3 2 7 2 7
## [1296] 8 3 3 2 7 2 7 7 7 7 2 2 3 7 7 3 7 7 3 2 8 2 3 3 1 1 3 2 3 7 1 2 7 7 7 3 3
## [1333] 1 1 8 7 7 7 7 8 1 1 1 7 2 7 2 2 8 2 3 7 2 7 3 3 8 7 1 3 2 3 3 8 1 2 3 3 2
## [1370] 1 3 7 7 3 2 7 8 2 2 2 3 1 7 8 2 7 7 7 7 7 7 2 3 1 1 3 7 7 7 7 7 3 2 3 3
## [1407] 3 7 7 1 3 7 7 2 7 7 2 3 1 3 1 3 1 2 2 3 3 2 2 8 8 2 7 8 2 2 3 3 2 2 7 3 7
## [1444] 3 8 3 2 3 1 3 1 2 7 2 6 7 7 8 3 7 3 3 3 2 7 3 3 3 7 7 7 7 3 7 1 2 3 2 2 3 7
## [1481] 2 7 3 3 7 7 2 7 7 3 7 7 2 7 7 2 3 2 2 3 8 3 3 7 7 2 3 7 7 7 8 8 3 1 1 3 2
## [1518] 3 3 2 2 2 2 8 2 2 3 3 2 7 3 3 3 2 3 7 7 7 7 2 2 2 3 7 2 3 1 3 3 1 7 2 2
## [1555] 7 7 3 2 7 8 2 7 1 3 2 3 3 1 7 7 7 2 7 7 3 3 2 3 7 7 7 8 7 3 7 7 2 7 7 7 7 2
## [1592] 2 3 7 3 3 8 3 3 3 1 7 3 2 2 7 2 3 7 3 3 3 2 3 7 7 7 7 7 8 3 7 2 6 2 2
## [1629] 2 1 3 3 3 3 3 3 2 3 7 7 7 7 3 7 6 7 3 3 1 3 7 7 2 7 7 3 7 2 3 3 3 3 1 2 3
## [1666] 3 3 3 3 3 3 7 3 7 1 1 7 7 3 3 3 1 7 7 3 1 1 3 3 2 2 2 2 3 7 7 7 7 2 2 3 6

```

```

## [1703] 7 2 1 1 2 3 7 3 2 8 2 1 3 3 2 2 7 8 3 3 3 3 3 3 3 2 7 7 8 3 3 3 3 2 3 7 7
## [1740] 2 8 3 3 3 2 7 1 3 3 3 3 7 8 7 7 8 2 1 3 3 3 2 3 7 7 8 7 1 1 3 1 3 2 7 8 7
## [1777] 7 8 2 7 7 7 7 7 2 8 7 7 7 2 7 6 7 3 7 8 7 2 2 3 3 7 1 4 3 6 2 2 2 1 3 2 2
## [1814] 3 3 7 7 7 1 1 1 3 3 3 1 3 3 3 2 3 8 7 3 3 2 2 2 3 3 3 3 7 3 1 7 2 7 3 3 7
## [1851] 2 2 2 7 2 7 2 1 7 7 3 7 3 1 3 3 2 2 3 7 2 3 3 7 3 3 3 3 1 7 2 7 2 3 7 3
## [1888] 1 3 3 7 7 3 3 7 3 6 3 1 3 2 7 3 3 3 8 1 1 7 7 7 2 3 2 3 3 3 2 7 2 7 3 8 3
## [1925] 7 7 3 3 7 7 3 3 7 2 2 8 7 1 2 1 3 3 3 1 3 3 7 7 3 7 8 7 2 2 7 7 7 7 7 2
## [1962] 2 7 2 7 7 7 2 7 7 3 2 1 3 3 7 2 2 3 2 3 7 7 2 7 7 3 7 2 7 7 1 2 3 3 1 1 7
## [1999] 7 2 2 2 7 2 1 2 2 3 3 3 3 3 7 7 2 3 7 7 2 6 7 7 2 2 7 3 3 7 2 7 7 3 3 7
## [2036] 7 7 7 3 1 1 2 2 2 3 7 7 7 2 1 1 8 2 2 7 2 6 7 7 7 2 3 8 3 7 3 8 7 7 3 1 2
## [2073] 3 7 2 7 2 2 2 2 3 3 2 3 7 2 2 7 7 7 7 3 2 7 2 8 7 7 3 2 3 3 3 7 7 3 1 7 3
## [2110] 7 2 3 7 2 7 3 3 2 7 2 1 3 7 8 2 2 3 3 3 7 8 2 7 1 2 3 1 1 7 8 7 3 2 3 2 3
## [2147] 1 2 7 2 3 2 7 3 1 2 7 7 3 1 3 7 7 3 3 1 3 1 3 3 3 3 1 3 7 8 2 8 2 7 2 6
## [2184] 2 8 3 7 2 7 8 7 3 7 7 7 2 3 7 3 3 2 2 3 7 3 3 1 3 7 2 8 2 7 3 8 7 3 3 3 7
## [2221] 2 3 8 3 3 1 3 3 3 3 7 2 2 3 2 7 8 2 1 3 2 3 7 7 7 7 3 7 2 2 2 7 7 7 7 7 7
## [2258] 2 7 7 7 7 7 7 7 7 3 7 7 3 2 2 2 3 3 3 2 7 7 3 8 7 2 7 2 3 7 2 2 7 7 2 7 7
## [2295] 7 7 2 7 6 7 7 7 7 2 7 3 3 2 3 3 2 3 3 7 7 2 7 2 7 7 2 7 7 2 8 3 3 7 3 3 7
## [2332] 2 2 7 2 7 2 7 7 3 2 2 7 3 8 2 1 2 2 3 7 7 2 2 2 7 3 2 1 3 2 7 1 7 7 7 3 7
## [2369] 2 3 7 1 3 8 3 7 1 3 2 3 3 2 7 8 3 3 3 7 3 7 2 8 7 7 7 7 2 7 2 2 2 2 2 8
## [2406] 2 8 2 3 8 3 2 7 3 2 7 2 3 6 7 7 3 3 2 1 3 8 1 1 1 8 7 2 3 7 7 8 1 3 3 2 3
## [2443] 2 2 2 7 7 2 7 2 2 2 3 7 8 8 7 2 7 7 7 3 2 3 7 7 7 2 2 7 8 3 3 3 2 2 2 3 7
## [2480] 3 3 2 3 2 2 7 7 7 7 2 7 2 6 2 2 8 7 7 3 3 3 1 1 7 7 6 8 1 3 2 2 2 1 3 3 3
## [2517] 8 7 2 2 7 7 7 7 8 3 1 1 3 3 2 7 2 7 2 7 3 3 3 2 3 2 3 7 7 7 3 3 6 8 3 2 7
## [2554] 7 2 7 1 3 2 2 7 3 3 7 7 7 2 6 8 3 3 3 2 2 7 2 7 7 7 8 7 7 8 7 7 3 2 7 7 3
## [2591] 1 1 7 8 7 7 7 7 7 2 7 7 3 7 1 7 3 3 2 3 3 1 3 3 2 7 8 7 3 7 2 7 5 6 2 7
## [2628] 3 1 1 7 7 3 3 3 1 7 7 3 7 2 7 2 3 8 7 7 7 2 7 2 3 3 7 7 3 7 2 7 2 6 7 7 3
## [2665] 3 1 3 3 3 7 7 2 7 2 7 2 3 1 1 2 2 7 7 7 8 3 1 1 2 3 1 3 3 2 3 3 1 4 3 3 3
## [2702] 3 3 7 2 7 7 7 2 6 1 3 8 3 2 7 7 7 1 3 2 2 2 3 3 3 1 2 7 2 7 7 7 7 2 7
## [2739] 7 7 7 2 7 2 3 3 3 1 3 8 3 7 7 2 6 7 7 7 2 7 2 1 1 3 3 6 1 2 3 2 3 7 3
## [2776] 7 2 3 8 3 7 8 7 3 3 3 7 1 2 1 7 7 7 8 3 2 3 7 7 2 2 3 7 7 2 3 7 7 2 3 8 2
## [2813] 7 2 7 2 7 7 2 2 3 3 3 3 8 3 1 3 2 2 3 2 3 2 7 7 7 2 1 7 2 2 3 3 8 7 1 2 6
## [2850] 1 3 7 7 7 7 2 2 3 7 7 6 2 3 7 2 7 7 2 3 3 3 3 7 8 7 3 2 7 7 7 2 7 1 7 3 2
## [2887] 7 7 2 2 7 7 7 2 8 7 2 2 3 3 7 2 2 2 3 7 7 2 7 7 3 2 7 3 7 7 7 3 2 3 2 7 1
## [2924] 7 7 7 7 7 1 7 3 3 3 3 7 2 2 2 3 7 6 7 8 7 2 7 8 2 3 2 3 3 3 3 7 7 1 3 7
## [2961] 7 8 3 3 2 7 1 3 3 2 1 3 3 2 3 3 2 7 2 3 3 3 3 3 1 7 7 8 7 3 2 2 7 2 3 2 3
## [2998] 3 3 1 3 1 2 3 7 7 2 3 7 2 1 4 1 1 3 3 8 1 2 7 7 7 2 3 8 7 3 3 2 3 7 7 3 3
## [3035] 2 2 7 2 7 3 2 7 7 6 1 2 7 2 7 7 7 7 7 6 7 8 7 3 3 8 7 7 8 8 2 2 2 3 6
## [3072] 7 2 7 3 3 1 3 8 7 3 7 7 3 1 3 1 1 2 2 2 2 7 3 3 8 1 3 3 3 3 2 7 1 2 3 2 7
## [3109] 7 7 8 7 3 2 2 2 2 7 2 3 3 3 1 1 3 3 3 7 7 3 2 3 7 3 1 3 3 3 3 1 7 1 2 2 2
## [3146] 2 3 2 2 7 6 4 3 2 7 3 3 7 2 3 1 7 2 7 8 3 7 2 8 8 2 1 2 3 2 1 3 7 7 7 2 7
## [3183] 7 6 7 1 3 2 7 3 3 7 1 1 2 7 7 1 3 2 3 3 8 7 7 8 2 8 2 8 3 1 8 3 3 2 7 3 7
## [3220] 2 7 7 3 3 3 2 3 3 3 7 2 1 1 3 3 3 7 3 2 2 1 2 8 2 3 7 2 7 3 3 2 3 3 3 3 3
## [3257] 3 3 1 1 3 7 2 7 3 3 3 7 7 7 8 2 6 7 2 2 7 7 2 6 3 3 7 1 7 7 3 2 2 7 7 7
## [3294] 7 1 6 1 3 6 1 3 2 3 1 3 3 3 1 3 2 2 2 2 7 7 3 1 8 7 2 3 2 7 7 1 1 3 2 8 3
## [3331] 3 3 3 1 7 3 7 3 8 7 2 7 3 7 8 7 2 2 1 3 7 3 3 7 3 7 7 8 1 3 7 7 7 7 1 3 3
## [3368] 3 7 3 2 2 7 7 2 7 7 8 1 3 8 3 7 2 6 3 3 3 1 3 3 2 3 1 2 7 7 7 7 2 7 7 1 1
## [3405] 7 3 3 1 3 3 8 3 7 7 3 7 7 2 3 7 7 8 2 3 3 7 8 3 3 2 7 7 8 3 2 1 1 1 3 2 3
## [3442] 8 7 6 3 2 7 7 3 2 7 7 3 7 3 3 3 3 1 1 3 3 2 3 3 7 7 8 2 3 7 3 3 7 7 2 2 8
## [3479] 7 2 2 7 2 7 7 7 3 3 3 2 2 3 3 7 2 2 3 2 2 7 7 8 3 3 2 3 1 3 3 3 3 2 8 3 3
## [3516] 7 8 3 3 1 3 2 7 8 2 7 7 8 3 3 2 2 3 7 2 7 7 7 7 2 2 3 3 3 3 8 1 2 7 7 8 3 2
## [3553] 8 1 7 2 2 3 3 3 7 3 1 1 3 7 7 6 1 7 6 6 3 3 7 8 3 3 7 7 7 2 6 7 7 7 7 7 3 7
## [3590] 7 8 3 7 1 2 3 3 2 3 3 8 3 2 2 7 7 7 3 3 3 3 3 3 2 3 1 3 3 2 3 8 2 7 8 7
## [3627] 7 7 6 3 2 7 8 3 2 3 3 3 2 3 2 1 3 7 7 2 7 7 1 2 2 2 7 7 8 3 3 3 3 3 2 7 8
## [3664] 7 2 1 3 2 8 2 3 3 3 1 3 3 7 7 7 7 7 7 8 2 2 7 1 2 2 1 3 2 3 2 2 7 7 2 7

```

```

## [3701] 7 3 2 7 7 2 3 1 1 2 3 2 7 7 7 2 2 7 2 2 7 7 8 1 3 3 7 7 7 2 7 2 7 7 7 7
## [3738] 7 7 7 7 7 8 3 7 2 7 7 7 3 2 3 7 3 2 2 2 2 7 7 7 2 7 7 1 3 3 2 8 3 2 3 7 3
## [3775] 3 3 2 2 3 7 7 3 3 3 3 1 2 3 3 2 2 1 2 7 7 3 7 3 3 2 3 3 3 3 3 2 1 1 2 3
## [3812] 3 7 2 2 3 7 7 2 3 3 7 2 2 3 8 8 6 7 7 7 7 7 3 3 3 3 3 3 8 7 3 3 7 7 2 2 7
## [3849] 2 2 2 7 1 7 2 1 2 3 2 1 3 3 3 7 1 7 2 8 3 2 7 7 3 3 1 3 3 1 3 3 3 3 2 3 3
## [3886] 1 7 7 7 7 7 2 8 2 1 2 3 3 3 3 3 8 2 3 1 3 3 3 3 1 3 7 7 1 3 3 1 3 7 7 3
## [3923] 2 2 7 7 7 2 2 3 2 3 3 2 2 3 3 3 3 2 7 3 2 2 7 3 3 3 3 3 2 3 3 7 7 7 1 3
## [3960] 3 2 2 7 7 3 3 7 1 1 8 8 7 7 2 8 2 7 7 3 3 7 7 3 6 7 3 6 2 2 7 3 3 3 2 7 2
## [3997] 2 7 2 2 8 2 3 3 3 7 7 2 3 8 3 3 7 7 7 2 2 7 8 7 3 3 2 2 3 3 2 3 1 7 7 2 2
## [4034] 7 1 2 3 2 7 2 3 3 8 2 2 1 1 7 8 7 7 3 3 8 3 2 3 2 2 3 3 3 3 8 3 2 3 8 8 1
## [4071] 1 3 7 2 3 3 7 7 7 3 8 1 1 3 1 3 3 3 8 7 7 3 3 6 7 7 2 2 5 7 1 1 8 7 2 3 1
## [4108] 7 3 7 7 7 7 3 3 2 3 7 2 8 3 2 7 3 7 3 2 2 1 7 1 3 3 8 7 2 2 2 1 3 2 7 8 2
## [4145] 7 3 7 1 3 3 3 2 3 3 1 1 3 1 3 7 7 7 7 2 8 7 7 3 2 3 2 7 7 7 7 2 7 2 8 1
## [4182] 7 7 7 7 2 1 1 3 7 5 2 7 3 7 2 7 3 1 7 3 2 2 2 7 7 7 6 3 3 7 7 7 7 7 7 8
## [4219] 6 7 2 2 7 7 1 1 7 3 7 1 1 3 2 1 3 8 1 2 7 7 7 3 7 7 7 7 2 2 2 2 2 3 3 3
## [4256] 1 1 1 3 7 2 7 7 1 8 1 2 1 2 2 8 7 8 7 3 1 7 5 7 2 7 7 7 7 3 3 3 3 1 3 7 8
## [4293] 7 3 1 7 7 6 3 2 3 2 3 2 7 2 3 3 2 3 3 7 3 7 7 3 7 7 2 3 7 1 3 1 3 3 7 2 8
## [4330] 3 2 7 2 2 3 3 3 7 7 3 7 7 2 2 7 7 7 7 1 8 3 3 3 7 1 1 3 3 1 3 7 7 7 2 7
## [4367] 8 3 2 2 7 2 7 2 2 2 7 2 7 7 7 3 1 3 3 2 7 3 2 3 7 7 3 2 3 1 3 3 3 2 7 2 3
## [4404] 7 3 3 7 7 7 7 2 7 7 7 7 3 2 8 3 1 2 3 7 7 2 2 7 7 6 7 7 7 2 7 7 1 2 1 3 3
## [4441] 7 2 2 2 2 2 7 2 7 7 7 2 7 2 7 3 3 2 3 3 3 3 1 3 3 3 8 7 3 7 3 3 3 3 3 3 3
## [4478] 7 7 2 2 7 8 7 3 7 7 1 8 1 3 3 3 2 2 8 2 1 7 7 2 2 2 2 7 7 3 7 7 3 7 7 1 3
## [4515] 7 7 2 2 7 8 7 7 3 2 2 7 2 7 3 1 7 7 3 7 7 7 3 7 7 7 2 1 7 8 3 2 1 3 1 3 7
## [4552] 3 3 7 7 7 7 1 3 3 7 3 2 3 2 3 1 3 1 7 7 7 3 2 7 2 7 3 7 8 2 3 8 7 2 2 2 7
## [4589] 7 2 7 3 3 7 3 7 8 7 2 7 8 7 3 3 3 2 7 7 3 2 8 7 3 2 1 8 3 7 7 6 3 2 3 3 3
## [4626] 3 2 7 2 7 7 2 7 2 7 7 7 7 6 7 1 3 3 3 7 7 7 8 2 2 7 7 2 7 3 3 3 3 2 3 2
## [4663] 3 3 1 3 7 7 1 2 2 2 2 7 7 3 3 7 8 7 3 8 8 3 1 1 2 8 1 3 3 3 2 3 7 7 7 8 7
## [4700] 3 3 3 1 1 2 7 7 2 7 7 7 1 2 1 3 3 7 7 2 2 7 1 7 7 1 3 2 3 2 3 2 7 7 7 7 7
## [4737] 7 7 7 3 7 7 7 7 7 7 7 7 2 3 3 3 3 1 3 1 1 2 3 3 3 3 7 7 7 2 3 3 3 3 7 8 1
## [4774] 2 3 7 7 7 2 7 3 1 2 2 2 1 3 3 8 2 1 3 3 7 7 7 1 2 3 2 3 1 1 3 8 7 8 7 3 7
## [4811] 7 3 3 3 7 3 3 3 3 7 1 2 1 1 3 3 3 7 7 3 2 7 7 7 2 7 3 1 2 3 7 1 2 7 3 3 2
## [4848] 8 3 7 2 8 2 7 7 7 3 7 8 2 6 2 3 3 7 6 8 7 7 3 2 3 1 3 1 2 3 3 1 7 2 6 7 2
## [4885] 7 2 7 2 7 2 7 7 7 7 7 2 3 3 1 7 7 2 7 3 7 2 1 3 3 2 3 1 1 7 3 2 3 7 7 1
## [4922] 7 7 2 7 7 3 7 7 2 7 7 3 1 3 3 2 2 3 2 8 7 7 7 7 3 7 7 2 7 7 7 7 2 7 7 1
## [4959] 8 3 7 3 7 7 1 3 7 7 7 7 3 3 3 3 7 8 2 3 2 3 1 3 7 7 2 2 3 2 1 3 3 4 8 8 2
## [4996] 3 3 1 3 2 7 2 7 7 7 2 6 7 1 2 1 1 1 2 2 2 3 7 7 7 7 7 7 7 7 3 7 7 8
## [5033] 7 7 2 7 7 2 3 7 2 7 7 8 2 2 3 2 1 3 7 3 7 2 3 8 2 2 1 7 8 7 7 7 7 3 2 2 7
## [5070] 7 7 2 7 7 3 2 2 3 7 3 7 3 3 3 7 1 2 3 1 3 3 3 7 1 1 3 1 1 3 3 3 3 2 3 3
## [5107] 7 8 2 2 2 3 1 3 3 3 7 7 2 8 2 7 7 7 8 2 6 7 7 7 8 8 3 2 3 2 3 7 7 2 7 2 8
## [5144] 3 7 7 7 3 3 7 7 2 3 7 7 7 3 3 1 3 3 8 7 3 7 7 7 7 7 6 7 7 3 1 3 3 3 7 3
## [5181] 7 2 7 7 2 2 8 7 6 2 3 7 2 2 7 7 7 2 6 3 3 2 2 7 7 7 7 7 2 7 7 7 7 7 2 2 7
## [5218] 7 2 2 7 2 2 2 7 2 2 3 7 3 2 7 2 2 3 7 7 7 7 3 7 1 7 3 3 3 1 7 3 2 7 7 3
## [5255] 3 3 7 2 2 3 3 1 3 3 3 7 7 7 7 7 2 7 2 1 7 7 2 2 7 7 7 7 7 8 1 1 7 3 2
## [5292] 2 3 3 8 3 7 8 7 7 6 6 2 3 1 2 7 7 3 2 4 3 3 7 7 8 3 3 3 1 1 1 3 7 2 2 7 7
## [5329] 3 3 7 2 7 7 7 7 3 2 2 2 7 2 3 2 7 2 3 3 7 2 8 3 3 8 2 7 2 7 7 7 8 2 2 7 3
## [5366] 3 3 7 7 3 7 7 3 2 3 2 7 2 3 7 2 2 8 3 3 8 8 3 7 7 3 2 7 2 2 2 2 8 1 2 7 7
## [5403] 3 3 2 7 7 7 3 3 7 2 7 2 1 3 3 7 2 3 3 3 3 2 7 6 2 2 7 3 3 3 1 3 6 7 3 1 3
## [5440] 3 3 3 7 8 7 3 2 2 7 2 7 7 2 2 3 3 2 7 2 8 3 7 1 3 2 1 3 3 3 7 2 2 3 1 7 7
## [5477] 2 7 7 1 3 7 7 7 3 3 1 3 3 3 3 3 7 7 3 1 3 7 7 7 2 7 3 3 3 3 7 7 2 3 3 3 3
## [5514] 7 7 7 6 8 3 3 2 3 2 7 8 7 7 7 2 3 6 2 7 7 7 7 7 7 3 1 3 1 1 1 2 7 3 7 2
## [5551] 3 2 3 2 3 7 7 7 7 2 7 7 6 7 3 3 2 3 3 3 3 1 3 3 3 2 7 7 2 8 7 2 7 7 7 7 7
## [5588] 2 3 7 7 7 7 1 7 7 7 7 7 3 2 3 3 2 2 3 3 2 7 7 3 7 8 7 2 2 2 3 7 2 7 7 8 7
## [5625] 7 7 6 2 3 8 7 2 7 2 2 2 7 7 2 3 8 7 7 7 2 2 8 7 2 2 7 2 7 7 1 3 3 3 1 3 3
## [5662] 3 8 3 1 3 3 3 7 8 7 7 3 3 3 7 2 7 1 2 7 7 7 7 2 2 3 2 2 2 3 8 2 7 2 7 3 1 1

```

```

## [5699] 3 2 3 2 3 3 3 7 3 7 7 7 6 7 3 7 2 2 7 2 7 2 2 3 3 7 3 8 2 7 7 7 7 7 8 7 7
## [5736] 3 7 7 3 7 7 3 7 7 2 2 3 3 3 2 1 7 2 2 7 1 6 3 1 7 3 7 7 8 2 3 2 2 7 7 7 2
## [5773] 7 3 2 3 3 7 7 3 3 7 7 7 1 2 1 2 2 1 2 3 7 7 3 8 7 7 7 3 3 3 3 3 3 3 3 3 3
## [5810] 3 7 7 3 7 3 7 3 3 8 7 7 7 2 3 3 7 7 7 7 7 7 3 2 1 1 2 3 2 2 7 3 2 7 2 2
## [5847] 3 7 7 7 7 7 7 7 2 3 1 1 3 7 3 7 7 7 7 7 2 3 1 2 2 7 7 2 7 3 2 7 2 7 3 8 8
## [5884] 7 6 3 7 7 7 7 3 3 3 7 7 7 2 7 7 7 8 7 7 7 2 7 7 3 7 7 3 7 7 2 2 7 7 6 2 7
## [5921] 7 7 7 3 2 8 8 7 7 1 3 3 3 3 3 2 3 3 3 3 8 7 3 7 2 7 7 2 7 7 2 3 2 3 1 2
## [5958] 2 7 3 7 8 7 7 7 2 3 7 7 3 7 2 7 7 8 2 7 7 7 7 2 2 7 7 2 7 2 6 2 3 2
## [5995] 3 2 3 3 3 7 7 3 1 1 7 7 7 7 7 7 6 7 3 2 6 7 2 7 7 2 7 7 3 7 8 3 2 2 3 7 3
## [6032] 7 2 7 1 1 7 2 7 7 3 3 3 3 2 3 7 1 1 2 2 2 3 3 1 1 8 7 2 3 7 7 2 7 7 2 8 8
## [6069] 7 3 3 3 3 3 3 7 2 7 7 7 8 1 1 8 3 3 1 7 7 1 7 7 8 7 7 7 7 3 7 6 7 6 7 7 3
## [6106] 3 1 2 3 1 2 3 2 8 3 2 6 7 3 3 8 1 3 3 2 7 3 2 3 7 7 8 2 3 7 3 1 2 3 7 2 7
## [6143] 7 2 7 7 7 3 3 2 2 8 7 3 3 3 3 7 8 2 3 3 3 1 3 3 3 3 7 7 3 7 2 7 7 6 2 7 7
## [6180] 7 2 7 7 8 2 8 7 2 2 3 8 7 2 1 2 3 3 2 7 7 6 3 2 8 3 2 3 3 3 6 2 7 7 7 7 7
## [6217] 8 7 2 7 2 8 3 2 3 1 3 3 3 3 7 3 7 8 6 2 3 3 7 7 3 8 1 7 2 7 7 7 7 7 7 1 8
## [6254] 7 8 3 1 3 7 7 3 7 1 7 2 3 8 7 2 3 8 2 2 7 7 2 2 7 3 3 3 3 3 2 2 3 3 3 2 2
## [6291] 8 7 3 7 3 6 2 7 3 7 1 3 2 3 7 2 3 3 3 8 3 7 7 7 7 7 7 2 2 3 3 7 7 2 7 7
## [6328] 7 2 2 7 7 7 2 7 2 7 2 7 7 6 3 7 7 7 7 2 3 7 7 1 3 1 7 7 2 2 7 7 2 7 2 2 3
## [6365] 8 7 7 7 2 1 3 7 7 7 3 2 3 7 7 2 8 1 3 1 3 8 1 7 7 2 7 7 7 2 7 7 3 3 2 8 2
## [6402] 7 7 7 3 3 2 2 2 3 3 3 3 7 2 1 3 3 3 3 7 7 7 7 2 8 7 6 1 3 3 3 2 1 3 2 2 7 2 3
## [6439] 3 2 2 7 7 2 7 2 1 3 3 7 7 7 3 3 7 2 7 2 7 2 2 7 7 2 7 2 2 6 2 2 7 2 8 3 1
## [6476] 1 3 3 3 2 2 3 3 3 2 7 2 1 8 2 3 3 2 3 3 3 7 8 8 2 2 2 1 7 3 3 3 2 3 7 2
## [6513] 7 2 3 3 2 8 7 3 6 7 2 3 3 1 1 6 7 3 1 2 7 2 6 6 3 3 7 2 2 7 3 7 1 3 3 2 3
## [6550] 2 2 7 2 3 3 3 1 3 8 2 7 1 7 3 7 2 1 2 3 3 3 3 7 7 8 8 3 2 2 7 7 7 2 7 2
## [6587] 2 7 7 7 3 7 2 3 2 1 3 1 3 7 7 8 2 3 7 2 3 2 3 2 3 7 7 2 7 3 2 7 2 2 6 7 2
## [6624] 8 3 3 6 7 8 1 3 6 7 7 8 3 2 7 2 8 2 7 8 3 2 3 1 3 1 3 2 1 3 2 3 3 3 2 3 8
## [6661] 7 7 2 2 2 2 2 7 7 2 7 3 2 7 2 7 2 7 2 7 7 7 7 2 3 1 2 2 7 2 7 2 8 7 7 3 3
## [6698] 7 7 3 3 3 1 3 7 3 2 7 2 3 8 3 1 1 3 2 2 7 7 3 2 7 7 2 7 7 3 2 2 1 3 3 3 3
## [6735] 3 7 7 8 3 3 3 3 3 2 2 2 2 3 7 7 7 7 7 2 7 2 7 3 7 7 2 3 3 3 3 1 2 2 2
## [6772] 1 3 3 2 7 3 8 7 7 7 8 3 3 3 2 3 7 2 3 2 7 7 3 3 1 1 1 1 3 1 3 3 3 7 7 3 3
## [6809] 1 1 1 3 1 8 1 7 2 6 1 1 3 1 2 8 7 7 5 7 2 7 7 7 3 3 7 2 7 7 6 2 1 7 7 2 2
## [6846] 3 7 3 3 7 6 2 7 7 7 7 7 7 2 8 8 3 3 3 2 7 3 6 7 8 3 7 1 3 7 6 1 3 6 2 6
## [6883] 7 2 6 3 3 7 3 8 2 7 7 7 7 2 1 2 1 2 6 3 1 2 7 7 3 7 1 7 7 7 3 3 1 3 2 7
## [6920] 1 2 7 2 7 2 8 7 7 7 2 7 3 3 3 3 1 3 7 3 2 2 7 2 7 2 7 3 3 2 3 3 3 3 2 7
## [6957] 2 7 7 7 7 7 2 7 2 7 7 8 7 2 3 3 3 1 3 3 3 7 8 2 7 7 2 7 2 7 7 2 2 1 7 2 7
## [6994] 2 2 7 1 3 7 7 7 3 2 3 3 7 1 1 3 3 2 7 7 3 7 7 7 7 2 3 3 3 8 2 8 2 3 7 2 7
## [7031] 7 7 7 3 3 3 3 7 7 7 3 7 2 3 3 3 3 7 3 3 2 7 7 7 2 3 8 2 7 7 3 3 3 3 7 2 2
## [7068] 1 8 7 3 3 3 2 1 3 2 1 8 3 3 3 1 7 8 1 3 7 7 3 7 7 7 2 2 2 3 2 8 2 3 3 1 7
## [7105] 2 3 8 7 7 7 7 7 2 3 6 3 7 3 7 3 1 2 7 7 7 7 7 2 7 3 2 7 6 7 7 7 1 3 1 3 7
## [7142] 2 2 3 1 3 3 7 7 3 7 1 3 1 2 2 7 7 7 3 3 7 2 8 3 7 7 3 8 3 7 7 2 6 3 1 3 1
## [7179] 1 7 7 3 2 8 3 3 3 1 1 3 3 1 8 2 7 7 2 7 7 7 3 1 2 1 7 7 8 3 3 3 7 8 3 7 2
## [7216] 7 7 3 7 3 1 7 3 3 2 7 2 2 2 7 7 7 8 7 3 7 1 2 2 2 2 7 6 2 2 3 1 8 7 7 3
## [7253] 7 7 2 7 7 2 2 2 7 2 3 3 2 3 1 3 3 8 3 7 3 1 2 2 8 3 7 2 6 1 1 2 3 7 7 7 3
## [7290] 8 3 8 1 2 7 3 2 2 1 1 2 1 2 3 3 3 7 3 7 7 2 1 1 8 3 7 3 3 1 2 2 3 3 7 7 2
## [7327] 3 7 2 6 1 3 1 3 3 7 3 7 7 7 7 8 7 2 1 3 3 3 1 2 2 2 8 2 2 8 3 3 2 3 7 3 3
## [7364] 3 8 1 3 2 1 3 7 8 7 2 7 3 3 2 3 7 2 2 7 7 7 2 8 7 3 2 3 2 7 2 3 7 2 8 3
## [7401] 7 2 7 2 7 3 3 2 8 3 7 3 3 3 2 3 2 1 2 1 3 1 3 3 3 1 7 7 7 7 7 3 1 3 2 2
## [7438] 2 2 2 3 7 3 3 3 1 2 2 3 2 1 2 3 3 3 7 7 2 8 2 7 8 7 8 7 8 7 7 7 1 3 6 3
## [7475] 6 7 7 7 2 7 7 7 3 2 7 2 2 6 3 7 2 2 3 2 1 3 7 7 7 7 7 3 7 3 7 1 3 7 3 8 2
## [7512] 2 2 2 7 2 7 2 7 7 6 1 1 3 2 2 2 7 2 3 3 3 3 3 2 7 1 7 3 1 1 3 2 2 2 2 2 1
## [7549] 1 1 7 2 3 7 1 2 2 7 7 7 2 3 3 3 3 7 3 3 8 7 7 2 2 7 7 7 7 3 7 6 7 3 7 6 2
## [7586] 7 7 7 7 2 8 3 3 3 2 7 3 3 3 2 1 2 8 7 1 1 8 3 7 2 7 7 7 7 1 7 2 3 2 2 3
## [7623] 7 7 2 8 2 3 2 2 2 2 8 7 7 7 7 7 2 7 2 1 1 1 2 3 1 3 2 8 3 3 3 7 7 7 7 7 7 1
## [7660] 2 1 1 2 2 7 7 6 7 2 7 7 7 7 2 3 7 3 7 7 6 7 2 7 6 8 7 8 7 8 7 7 7 7 2 3 2 2

```

```

## [7697] 1 7 7 7 2 3 7 7 7 7 7 7 3 1 2 3 3 3 2 3 2 1 7 7 7 3 7 3 3 3 3 7 3 3 3 3 3
## [7734] 3 7 7 3 8 2 7 7 3 7 7 7 7 3 2 7 7 7 7 3 3 2 2 7 7 3 3 3 3 8 7 6 2 3 2 3
## [7771] 7 1 4 3 7 7 7 7 3 7 7 1 3 8 3 7 1 3 1 8 1 1 3 2 7 3 2 7 7 3 7 3 3 3 3 1
## [7808] 1 3 3 7 7 3 2 3 3 7 2 6 8 2 8 7 7 2 7 3 3 3 7 3 3 3 2 3 3 3 7 7 7 2 7 7 8
## [7845] 3 2 6 1 3 3 3 3 1 7 7 7 3 2 7 7 1 7 7 3 3 3 1 7 7 3 3 7 2 7 7 2 3 3 1 3 2
## [7882] 2 8 3 2 3 2 3 2 8 7 2 7 3 2 7 8 2 1 2 3 3 2 3 3 3 1 7 3 8 7 3 1 3 6 7 3 3
## [7919] 3 7 7 7 2 1 7 2 7 7 7 2 7 3 2 7 3 7 3 6 7 8 3 8 1 2 3 1 1 2 2 1 3 3 2 7 7
## [7956] 3 2 7 2 3 3 2 1 3 3 8 7 1 3 3 3 3 3 1 3 1 3 7 7 3 3 7 2 2 3 2 2 7 7 7 7 1
## [7993] 3 1 1 1 2 8 2 7 7 2 2 7 7 7 8 7 7 2 2 2 2 8 7 7 2 8 3 8 7 1 2 2 1 7 7 7 2
## [8030] 2 1 2 3 1 1 3 2 7 3 2 2 7 7 7 8 2 1 3 2 1 3 2 7 1 2 3 8 2 3 7 2 3 7 7 7 7
## [8067] 3 7 2 2 1 3 3 1 8 1 7 7 7 8 2 3 2 1 8 2 3 3 3 8 2 7 2 7 7 7 2 2 7 7 6 7 6
## [8104] 2 2 3 2 3 3 2 3 2 1 3 2 7 7 2 3 7 3 8 3 3 3 1 3 3 1 7 7 7 7 1 7 2 1 8 8 8
## [8141] 3 7 7 3 3 2 7 7 2 7 7 7 7 5 7 7 3 2 3 2 7 2 7 2 2 3 3 2 6 3 7 2 7 2 7 3 2
## [8178] 2 7 1 8 3 1 1 3 3 3 8 7 7 2 7 1 3 3 7 3 2 3 1 3 2 7 7 6 7 3 3 1 3 3 1 3 1
## [8215] 3 1 1 3 1 3 1 3 3 2 7 8 7 7 7 7 7 2 2 7 3 1 6 3 3 3 3 3 7 8 7 7 1 3 3 1 1
## [8252] 1 3 7 7 8 7 2 2 8 7 7 7 1 7 2 7 3 3 6 3 6 3 3 2 8 1 1 3 7 3 3 3 3 8 1 2 2
## [8289] 2 2 3 3 1 3 7 8 7 3 3 7 7 6 3 3 3 2 2 7 8 3 3 7 6 3 3 2 3 3 8 7 3 1 7 2 2
## [8326] 3 1 1 7 7 7 7 2 7 7 8 7 7 2 2 2 7 2 2 3 3 2 7 8 2 7 3 1 3 1 3 7 2 3 2 3 1
## [8363] 2 3 3 3 7 7 2 3 3 1 7 7 2 2 3 3 3 2 3 3 3 7 7 3 7 2 7 2 7 7 7 2 7 3 2 1 3
## [8400] 3 2 7 7 7 7 2 7 2 2 3 3 2 3 8 3 7 2 7 7 7 7 7 2 7 6 1 3 7 2 2 7 8 3 3 1 1
## [8437] 8 3 3 3 3 3 3 1 3 3 7 7 3 8 7 2 3 3 1 3 3 1 1 1 2 7 7 3 3 7 7 3 6 3 3 3 3
## [8474] 3 1 3 3 2 7 7 7 3 7 2 8 7 7 7 5 8 2 7 8 3 1 3 7 7 7 2 7 3 3 7 3 1 2 3 8 7
## [8511] 7 3 1 3 7 8 7 1 7 7 7 8 7 2 3 2 1 3 3 3 7 7 2 7 7 7 1 2 3 3 2 3 2 7 7 7 7
## [8548] 7 7 3 3 3 3 8 2 2 8 3 8 3 3 1 6 1 2 3 3 2 2 8 3 3 3 7 7 3 7 7 7 7 1 3 3 2
## [8585] 3 1 2 2 3 3 7 3 3 3 3 7 7 3 3 7 7 3 2 3 3 2 7 7 7 3 7 2 7 7 7 3 7 3 2 7 1
## [8622] 3 3 1 2 3 7 2 3 3 7 1 1 3 1 2 2 2 2 7 1 1 8 3 1 3 3 3 2 1 2 7 7 7 7 7 7
## [8659] 1 2 2 1 2 2 2 2 2 3 3 7 3 3 3 3 3 2 1 2 1 2 6 2 7 7 2 2 7 2 7 3 3 3 2 2 7
## [8696] 3 3 3 3 6 8 3 2 3 8 7 7 7 3 3 7 3 3 2 2 1 1 1 8 3 3 8 1 2 2 7 3 3 3 3 2 7
## [8733] 7 7 2 7 7 1 3 3 2 7 3 3 7 7 7 3 3 6 7 3 3 2 1 1 7 8 7 7 6 8 7 7 3 1 1 1 3
## [8770] 3 3 2 3 2 3 3 2 8 3 3 3 2 3 7 3 2 3 3 3 8 2 7 2 2 3 7 7 2 6 3 7 7 7 7 3
## [8807] 7 1 3 2 7 7 2 3 7 2 7 7 2 3 3 1 3 3 3 3 2 2 7 7 8 7 3 7 7 7 7 2 7 3 7 7 7
## [8844] 2 1 3 2 3 3 8 1 3 7 2 3 3 2 3 6 3 7 7 7 7 2 7 2 3 3 2 8 7 1 1 1 3 1 7 7 8
## [8881] 7 2 3 3 1 1 1 7 8 7 6 7 6 3 7 7 3 7 7 7 1 3 3 3 7 7 8 2 8 7 3 8 1 3 2 1 1
## [8918] 3 1 7 7 3 3 1 2 1 1 2 3 3 3 2 7 7 3 8 2 3 2 3 3 1 1 3 7 7 7 2 7 2 3 3 2 2
## [8955] 3 7 3 1 2 1 2 3 7 7 2 1 3 2 3 8 3 7 3 2 2 3 3 1 7 1 1 1 1 1 3 1 3 3 7 6
## [8992] 2 3 1 2 2 3 3 2 2 1 7 7 2 8 2 7 7 7 3 7 3 2 2 2 3 3 7 3 7 2 8 3 7 7 7 7 7
## [9029] 7 3 7 7 7 6 1 3 7 7 3 5 3 3 7 7 3 2 2 2 7 7 7 7 3 3 7 7 7 6 2 3 2 3 7 2 8 1
## [9066] 3 7 7 3 7 3 7 7 2 2 1 3 7 7 7 2 7 7 7 2 3 1 2 3 2 8 3 1 7 2 7 2 2 2 3 3 3
## [9103] 3 1 1 3 3 3 2 3 2 7 7 7 2 3 3 3 3 2 2 2 7 1 1 1 7 3 1 1 3 7 7 2 3 3 3 1 2
## [9140] 2 7 7 3 8 2 1 7 2 8 7 2 3 3 2 3 2 7 8 8 1 1 2 7 3 1 6 7 7 8 7 7 7 7 7 3 7
## [9177] 1 7 7 3 3 3 7 7 7 3 6 3 3 2 1 3 1 3 2 3 7 2 2 3 7 7 3 2 8 7 8 3 7 7 7 7 7
## [9214] 7 2 2 1 7 2 1 7 6 7 1 1 1 3 2 7 3 8 1 8 7 2 2 3 3 3 2 1 3 2 3 7 2 7 7 3 2
## [9251] 8 3 2 7 7 7 7 7 7 7 3 7 2 1 2 7 7 7 2 6 7 2 6 2 2 3 3 1 3 7 2 3 3 7 3 1
## [9288] 3 2 7 7 7 1 2 3 1 3 3 3 1 3 3 3 3 1 2 7 7 7 2 8 3 3 2 7 3 3 3 7 7 7 1 1 8
## [9325] 3 3 7 3 7 7 7 3 7 7 3 3 2 3 2 7 2 3 2 3 1 3 3 2 8 7 3 3 3 3 7 7 7 2 3 7 2
## [9362] 2 2 7 7 7 3 2 7 7 3 3 2 2 2 7 7 7 3 3 3 3 3 2 7 8 7 7 7 7 2 3 8 2 7 3 2 7
## [9399] 6 2 7 3 3 6 2 3 3 8 7 1 2 3 6 1 3 7 7 3 3 1 1 2 2 3 7 6 2 2 8 2 3 3 3 2 3
## [9436] 3 1 7 3 3 7 2 3 3 3 7 2 2 7 3 8 7 2 7 3 3 3 1 7 3 7 2 7 3 2 7 2 3 7 7 7 3
## [9473] 7 2 3 2 1 1 2 1 1 3 3 2 8 8 3 7 3 2 6 7 8 6 7 2 7 7 7 8 7 7 2 7 7 2 7
## [9510] 2 8 1 3 6 2 8 3 3 3 7 7 3 3 3 2 2 3 2 3 3 7 3 8 2 2 3 7 3 3 2 7 7 8 7 3 7 2 6 7 3 3 2
## [9547] 7 3 1 3 3 2 7 2 2 3 2 1 2 2 3 3 1 3 7 2 1 3 7 1 3 3 3 2 7 3 3 2 7 7 2 2 2 2
## [9584] 3 7 3 2 3 3 3 1 1 1 7 7 7 3 3 7 3 8 2 2 3 7 3 3 2 7 7 8 7 3 7 2 6 7 3 3 2
## [9621] 2 3 3 1 3 3 3 7 2 1 3 2 8 7 2 8 7 2 7 4 2 2 2 3 2 3 1 7 7 6 8 7 2 3 1 7 3
## [9658] 7 2 2 6 7 3 7 7 7 7 2 7 7 2 2 3 1 3 7 2 7 7 7 8 7 2 8 7 7 2 1 2 7 7 3 8 2

```

```

## [9695] 7 3 2 3 7 7 3 7 7 2 7 3 3 7 7 7 3 1 2 2 3 3 3 2 2 3 7 7 7 8 3 7 2 2 7 3 2
## [9732] 8 8 7 7 3 7 7 2 2 7 6 2 7 3 3 3 1 3 7 3 3 3 7 7 7 7 2 7 2 7 7 7 7 2 7 7 2
## [9769] 2 2 3 3 2 7 4 1 1 8 1 1 7 3 2 7 7 7 2 2 3 3 3 2 3 3 1 1 3 3 3 3 3 2 3 7 7
## [9806] 3 1 3 1 2 3 7 7 2 2 2 3 2 3 3 8 1 7 7 3 3 3 7 3 3 3 1 3 2 1 1 2 7 7 3 8 7
## [9843] 3 3 1 3 2 2 2 7 7 3 7 2 7 7 2 6 2 7 7 7 7 2 7 3 1 1 3 3 7 3 7 3 1 3
## [9880] 2 3 7 8 7 7 7 7 7 7 7 2 2 7 2 8 3 3 3 3 3 3 1 3 2 3 3 2 7 7 2 3 7 7 2
## [9917] 7 7 7 2 1 1 3 3 7 8 7 7 7 6 3 2 2 7 3 3 7 3 2 7 2 8 8 2 7 3 3 6 6 7 7 7 3
## [9954] 7 3 3 2 7 7 3 7 3 3 3 7 7 2 2 7 2 7 2 3 2 2 7 3 2 3 8 7 2 2 7 8 7 7 7 2 3
## [9991] 7 3 7 7
##
## Within cluster sum of squares by cluster:
## [1] 1692.9327 1850.0638 1220.0787 1460.2766 778.7336 1798.3840 994.1753
## [8] 958.1986
##   (between_SS / total_SS =  73.1 %)
##
## Available components:
##
## [1] "cluster"      "centers"       "totss"         "withinss"      "tot.withinss"
## [6] "betweenss"    "size"          "iter"          "ifault"

```

The within cluster sum of squares by cluster value is 73.1% for k= 8, which is not very different from the Within cluster sum of square by cluster value for k = 7 (70.2%).

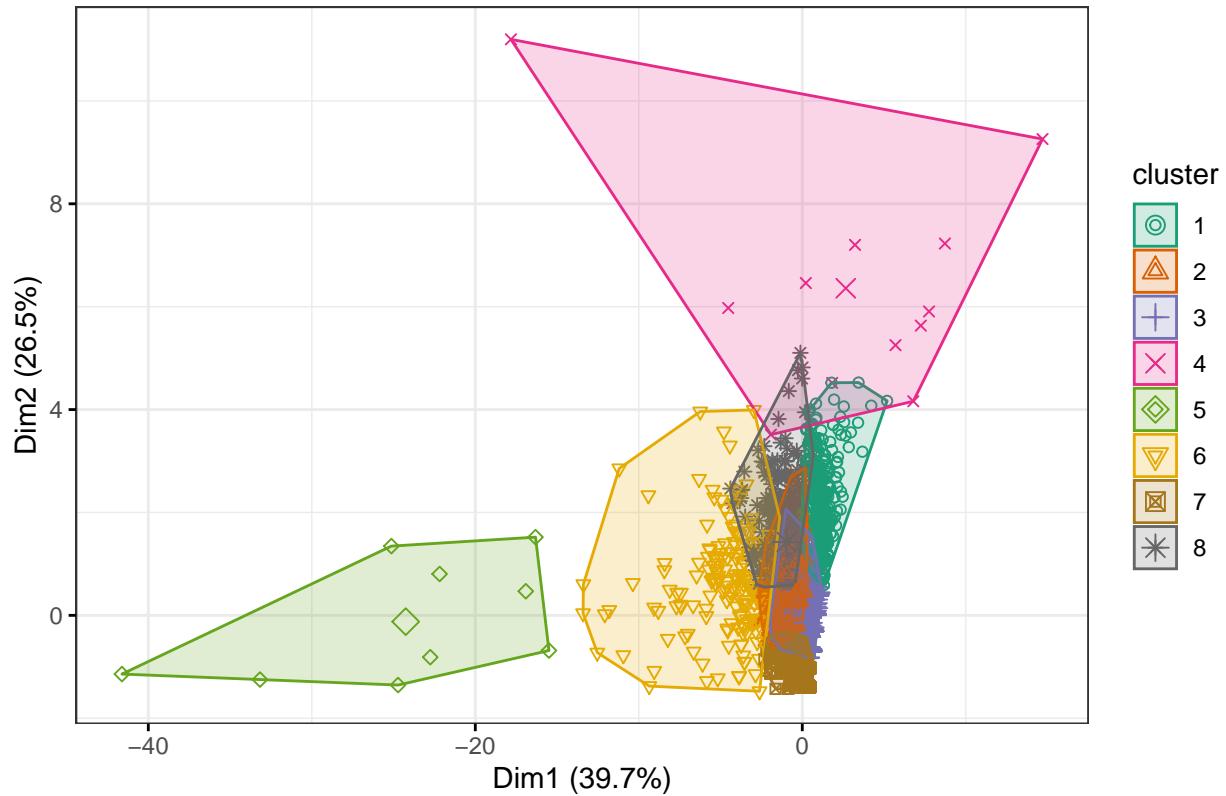
Let's plot the K-means clusters

```

library(RColorBrewer)
fviz_cluster(clustering_results_8, data = data1[, (15:18)],
             palette = brewer.pal(n = 8, name = "Dark2"),
             geom = "point",
             ellipse.type = "convex",
             ggtheme = theme_bw()
)

```

Cluster plot



There are still some overlaps between cluster groups.

Compute PCA and extract individual components and extract individual components.

```
# Dimension reduction using PCA
results_pca_8 <- prcomp(data1[, (15:18)], scale = TRUE)

# Coordinates of individuals
ind.coord <- as.data.frame(get_pca_ind(results_pca_8)$coord)

# Add clusters obtained using the K-means algorithm
ind.coord$cluster <- factor(clustering_results_8$cluster)

# Add Region groups from the original data sett
ind.coord$Region <- data1$Region

#look at the first few rows to double check
head(ind.coord)
```

	Dim.1	Dim.2	Dim.3	Dim.4	cluster	Region
## 1	-0.0426607	-1.0508481	0.14635987	0.31244832	7	South
## 2	-1.1546701	-0.7511273	0.25351937	0.22331213	7	South
## 3	0.3107253	-1.1081474	-0.01276717	0.15000936	7	West
## 4	0.6283963	1.9637213	0.37261495	1.54367992	1	South
## 5	0.5768386	-0.4056909	0.55980463	-0.06649362	3	South
## 6	-0.3936310	0.3152509	-1.56442060	-0.20148637	2	West

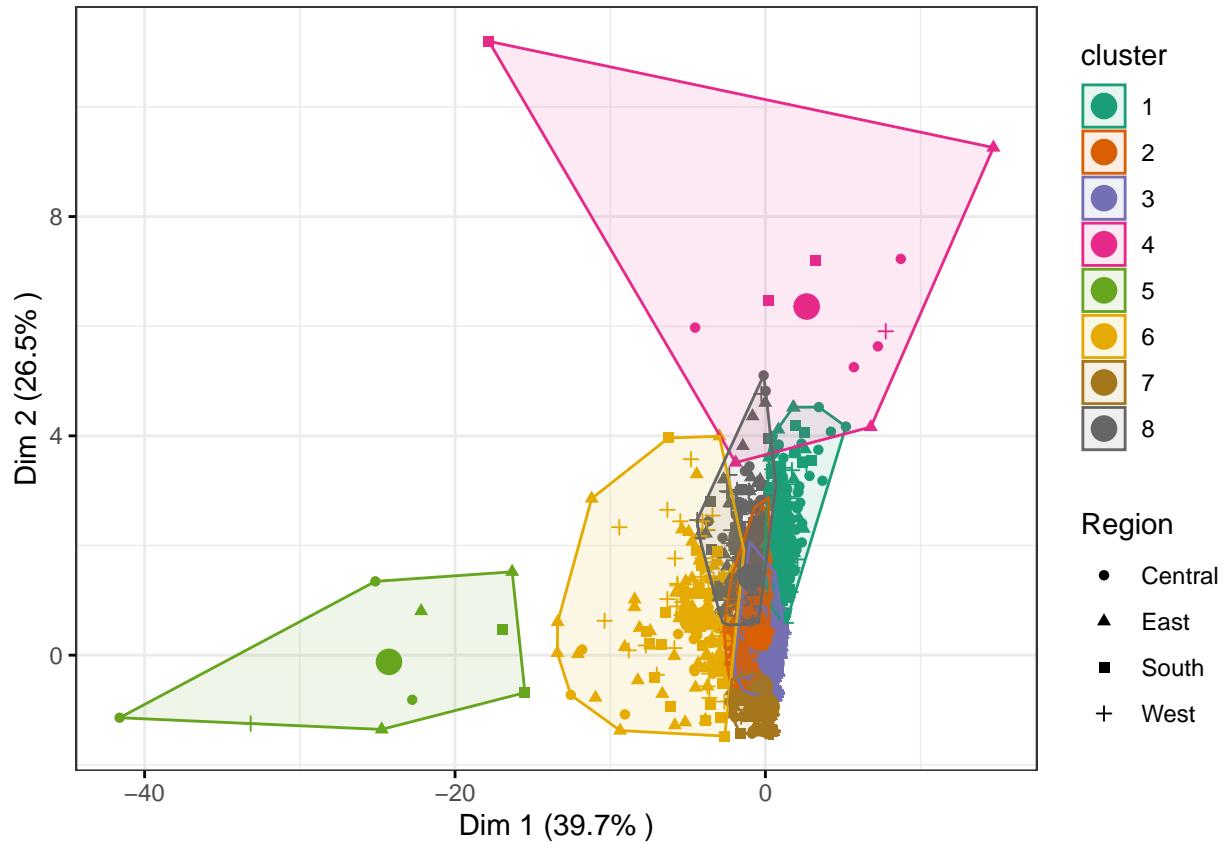
```

# Percentage of variance explained by dimensions
eigenvalue <- round(get_eigenvalue(results_pca_8), 1)
variance.percent <- eigenvalue$variance.percent
head(eigenvalue)

##          eigenvalue variance.percent cumulative.variance.percent
## Dim.1      1.6           39.7                  39.7
## Dim.2      1.1           26.5                  66.2
## Dim.3      0.9           22.0                  88.3
## Dim.4      0.5           11.7                 100.0

ggscatter(
  ind.coord, x = "Dim.1", y = "Dim.2",
  color = "cluster", palette = brewer.pal(n = 8, name = "Dark2"), ellipse = TRUE, ellipse.type = "convex",
  shape = "Region", size = 1.5, legend = "right", ggtheme = theme_bw(), #change point size
  xlab = paste0("Dim 1 (", variance.percent[1], "%)" ), ylab = paste0("Dim 2 (", variance.percent[2], "%)" )
) +
  stat_mean(aes(color = cluster), size = 4)      #add cluster centroid using stat_mean()

```



Clustering Validation

(<https://towardsdatascience.com/clustering-analysis-in-r-using-k-means-73eca4fb7967>)

Silhouette coefficient can be used to evaluate the goodness of the clustering. First, for each observation i, it calculates the dissimilarity between i and all the other points within the same cluster. This value is called average dissimilarity D_i .

Then calculate the dissimilarity between i and all the other clusters and get the lowest value among them. Find the dissimilarity between i and the next closest cluster, called C_i .

Next find the silhouette width which is the difference between C_i and D_i , divided by the maximum difference between C_i and D_i . $S_i = (C_i - D_i) / \max(D_i, C_i)$

$S_i > 0$ means the observation is well clustered. The closer it is to 1 the better it is clustered.

$S_i < 0$ means the observation is wrongly clustered.

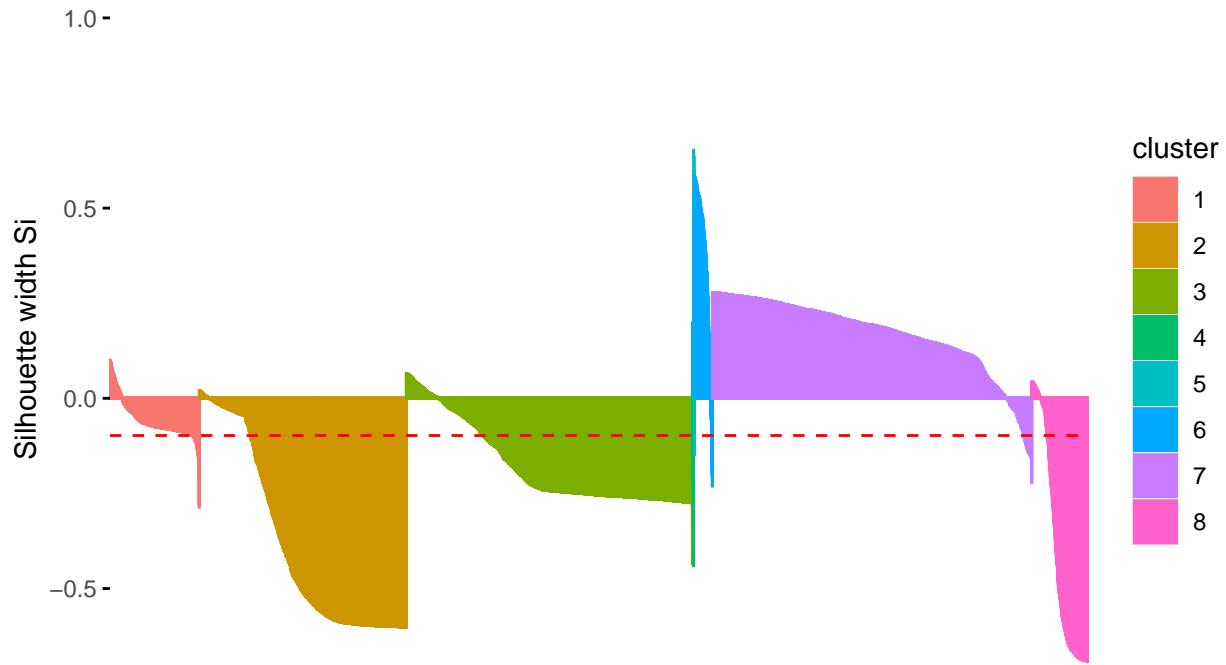
$S_i = 0$ means the observation is between 2 clusters.

```
library(cluster)
library(factoextra)

sil <- silhouette(clustering_results_8$cluster, dist(data1[, (15:18)]))
fviz_silhouette(sil)

##   cluster size ave.sil.width
## 1       1  912      -0.06
## 2       2 2114      -0.38
## 3       3 2923      -0.18
## 4       4   12      -0.17
## 5       5    9       0.48
## 6       6  182       0.37
## 7       7 3263       0.17
## 8       8  579      -0.40
```

Clusters silhouette plot
Average silhouette width: -0.1



From the table, 5 of the 8 clusters have a negative silhouette width which means that some observations may be in the wrong cluster, so the clustering is not very good.

Method 2: K-medoids

Following this tutorial: <https://towardsdatascience.com/clustering-on-mixed-type-data-8bbd0a2569c3>

K-means algorithm is limited in that it can only work with numerical data, whereas our dataset contains both numeric and categorical data.

We will now try the PAM clustering algorithm (Partitioning across medoids). K-medoids is more robust to outliers and noise than K-means. This algorithm uses Gower distance to measure the partial dissimilarity across individuals, and ranges in [0 1]. Standardization is first applied to the features, and the distance between individuals represents the average of all feature specific distances.

Partial dissimilarity is different depending on the type of variable: numeric or categorical.

Numeric features - partial dissimilarity is dependent on absolute difference between 2 observations (x_i and x_j), and the maximum range observed from all individuals. $d_{ij}^f = |x_i - x_j| / |\max_N(x) - \min_N(x)|$ where N is the number of individuals in a dataset.

Categorical/Qualitative features - feature dissimilarity is equal to 1 if y_i and y_j do not have the same values, otherwise it is 0.

One method to determine the number of clusters is by using the Silhouette coefficient.

Data Preparation

For this K-medoids analysis we will omit the Region variable because there are only 4 values and the information is too broad.

We will try to cluster transactions according to the following features:

```
data2 <- data1 %>%
  select(Ship_Mode, Segment, City, State, Sub_Category, diff_in_days, Sales, Quantity, Disc...  
  
#convert all character data type to factor:  
data2[sapply(data2, is.character)] <- lapply(data2[sapply(data2, is.character)],  
  as.factor)
```

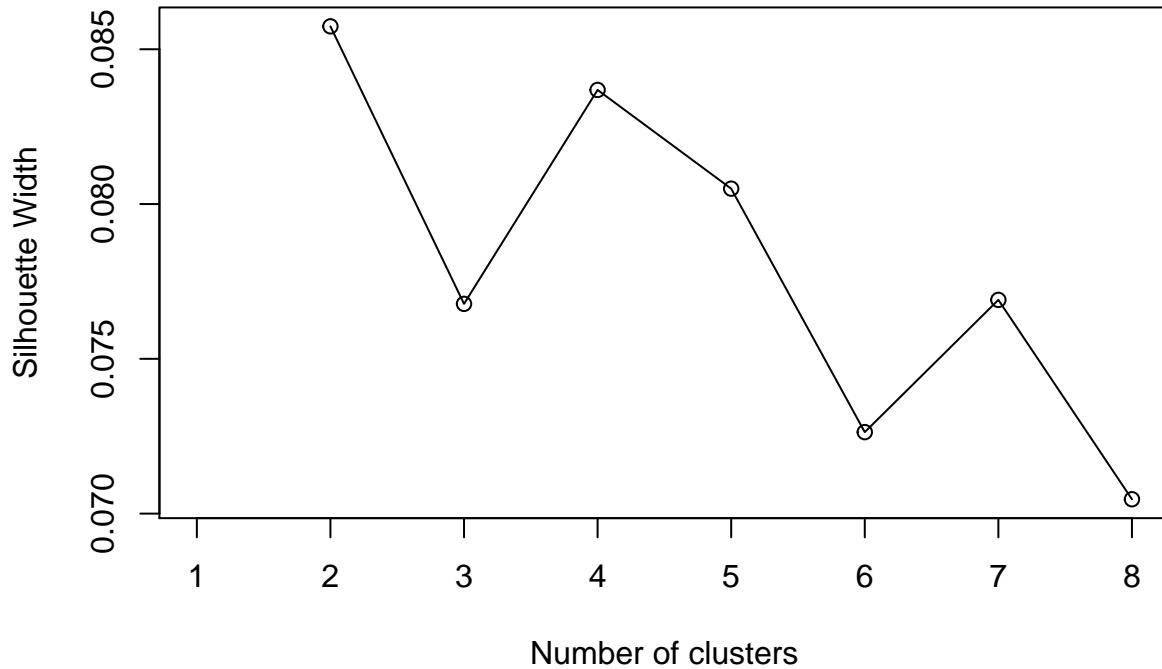
Compute Gower distance

```
gower_dist <- daisy(data2, metric = "gower")  
  
gower_mat <- as.matrix(gower_dist)  
  
#Print most similar transactions  
data2[which(gower_mat == min(gower_mat[gower_mat != min(gower_mat)])), arr.ind = TRUE][1, ], ]  
  
##          Ship_Mode    Segment      City State Sub_Category diff_in_days Sales
## 4875 Standard Class Consumer Houston Texas      Binders        4 1.188
## 3326 Standard Class Consumer Houston Texas      Binders        4 1.234
##          Quantity Discount Profit
## 4875       1     0.8 -1.9602
## 3326       1     0.8 -1.9744  
  
#Print most dissimilar transactions  
data2[which(gower_mat == max(gower_mat[gower_mat != max(gower_mat)])), arr.ind = TRUE][1, ], ]  
  
##          Ship_Mode    Segment      City State Sub_Category diff_in_days
## 6827 Standard Class Corporate Lafayette Indiana      Copiers        7
## 6798 Same Day Consumer Houston Texas      Binders        0
##          Sales Quantity Discount Profit
## 6827 17499.950       5     0.0 8399.9760
## 6798   3.798       1     0.8 -6.0768
```

Try to figure out the number of clusters to use by using the silhouette coefficient. Typically the number of clusters used is between 2 and 8.

```
sil_width <- c(NA)
for (i in 2:8){
  pam_fit <- pam(gower_dist, diss = TRUE, k = i )
  sil_width[i] <- pam_fit$silinfo$avg.width
}

plot(1:8, sil_width,
  xlab = "Number of clusters",
  ylab = "Silhouette Width")
lines(1:8, sil_width)
```



2 clusters has the highest silhouette width, while 4 has the second highest. 2 Clusters may be too simple, so We will pick $k = 4$.

To look at a summary of each of the clusters:

```

k <- 4
pam_fit <- pam(gower_dist, diss = TRUE, k)
pam_results <- data2 %>%
  mutate(cluster = pam_fit$clustering) %>%
  group_by(cluster) %>%
  do(the_summary = summary(.))
pam_results$the_summary

```

```

## [[1]]
##           Ship_Mode      Segment          City        State
## First Class : 528 Consumer :1586 San Francisco: 279 California:638
## Same Day     : 284 Corporate : 386 Seattle       : 113 Texas      :195
## Second Class:1262 Home Office: 200 Los Angeles   : 100 Washington:142
## Standard Class: 98 Houston      :  85 Ohio       :127
##                         Chicago     :  66 Illinois    : 95
##                         Columbus    :  53 New York   : 90
##                         (Other)    :1476 (Other)    :885
##           Sub_Category diff_in_days      Sales      Quantity
## Furnishings:407 Min.    :0.000 Min.    : 0.99 Min.    : 1.000
## Paper       :260  1st Qu.:2.000 1st Qu.: 18.58 1st Qu.: 2.000
## Accessories:196 Median   :2.000 Median   : 54.79 Median   : 3.000
## Storage     :193 Mean     :2.358 Mean     : 216.38 Mean     : 3.699

```

```

## Art :188 3rd Qu.:3.000 3rd Qu.: 197.91 3rd Qu.: 5.000
## Phones :168 Max. :7.000 Max. :13999.96 Max. :14.000
## (Other) :760

##      Discount          Profit           cluster
## Min. :0.0000  Min. :-1862.312  Min. :1
## 1st Qu.:0.0000 1st Qu.: 2.936  1st Qu.:1
## Median :0.0000 Median : 9.752  Median :1
## Mean :0.1113  Mean : 32.005  Mean :1
## 3rd Qu.:0.2000 3rd Qu.: 31.693 3rd Qu.:1
## Max. :0.8000  Max. : 6719.981 Max. :1
##
##
## [[2]]
##             Ship_Mode          Segment            City
## First Class : 309 Consumer :2746 Los Angeles : 615
## Same Day     :  69 Corporate : 475 San Francisco: 165
## Second Class : 159 Home Office: 237 Houston       : 152
## Standard Class:2921
##                                         Chicago       : 136
##                                         Seattle       : 135
##                                         San Diego     : 100
##                                         (Other)      :2155
##             State          Sub_Category diff_in_days      Sales
## California:1142 Binders :1063 Min. :0.000  Min. : 0.444
## Texas      : 408 Phones   : 312 1st Qu.:4.000  1st Qu.: 14.943
## Illinois   : 221 Art      : 263 Median :5.000  Median : 49.564
## Washington: 164 Storage  : 263 Mean   :4.711  Mean   : 225.913
## Ohio       : 161 Accessories: 231 3rd Qu.:6.000  3rd Qu.: 217.584
## Florida    : 135 Chairs   : 213 Max.   :7.000  Max.   :10499.970
## (Other)    :1227 (Other)  :1113
##      Quantity          Discount          Profit           cluster
## Min. : 1.000  Min. :0.0000  Min. :-6599.978  Min. :2
## 1st Qu.: 2.000 1st Qu.:0.0000 1st Qu.: -0.936  1st Qu.:2
## Median : 3.000 Median :0.2000 Median : 6.774  Median :2
## Mean   : 3.846 Mean   :0.2148 Mean   : 19.673 Mean   :2
## 3rd Qu.: 5.000 3rd Qu.:0.2000 3rd Qu.: 25.192  3rd Qu.:2
## Max.  :14.000 Max.  :0.8000 Max.  : 5039.986 Max.  :2
##
##
## [[3]]
##             Ship_Mode          Segment            City          State
## First Class : 281 Consumer : 295 Philadelphia:506 Pennsylvania:537
## Same Day     :  83 Corporate : 190 Seattle       : 79 Texas       :147
## Second Class : 204 Home Office:1188 Houston       : 66 Illinois     : 90
## Standard Class:1105
##                                         Chicago       : 55 Washington   : 89
##                                         Columbus     : 34 Ohio         : 85
##                                         Springfield : 31 California   : 72
##                                         (Other)      :902 (Other)     :653
##             Sub_Category diff_in_days      Sales           Quantity
## Paper      :322 Min. :0.000  Min. : 0.852  Min. : 1.000
## Binders    :187 1st Qu.:3.000 1st Qu.: 15.552 1st Qu.: 2.000
## Phones     :163 Median :4.000  Median : 47.984 Median : 3.000
## Storage    :141 Mean   :4.014  Mean   : 216.941 Mean   : 3.727
## Accessories:137 3rd Qu.:5.000 3rd Qu.: 203.976 3rd Qu.: 5.000
## Furnishings:133 Max.  :7.000  Max.  :22638.480 Max.  :14.000

```

```

## (Other)      :590
##   Discount       Profit       cluster
##   Min.    :0.0000  Min.   :-3399.980  Min.   :3
## 1st Qu.:0.0000  1st Qu.: -2.131  1st Qu.:3
## Median  :0.2000  Median  :  5.443  Median  :3
## Mean    :0.2065  Mean    : 10.078  Mean    :3
## 3rd Qu.:0.2000  3rd Qu.: 20.539  3rd Qu.:3
## Max.    :0.8000  Max.    : 2591.957  Max.    :3
##
##
## [[4]]
##           Ship_Mode        Segment          City
## First Class   : 420  Consumer   : 564  New York City: 789
## Same Day     : 107  Corporate  :1969  Seattle      : 101
## Second Class  : 320  Home Office: 158  Houston      :  74
## Standard Class:1844
##                               Columbus    :  70
##                               Springfield :  59
##                               Chicago     :  57
##                               (Other)     :1541
##           State        Sub_Category  diff_in_days   Sales
## New York   : 925  Paper       :600  Min.   :0.000  Min.   : 1.24
## Texas      : 235  Storage     :249  1st Qu.:4.000  1st Qu.: 20.34
## California: 149  Phones      :246  Median  :4.000  Median  : 62.28
## Washington: 111  Art         :215  Mean    :4.248  Mean    : 253.84
## Ohio       :  96  Accessories:211  3rd Qu.:5.000  3rd Qu.: 219.10
## Florida    :  94  Furnishings:205  Max.    :7.000  Max.    :17499.95
## (Other)    :1081  (Other)     :965
##           Quantity      Discount       Profit       cluster
##   Min.    : 1.000  Min.   :0.00000  Min.   :-3839.990  Min.   :4
## 1st Qu.: 2.000  1st Qu.:0.00000  1st Qu.:  3.972  1st Qu.:4
## Median  : 3.000  Median :0.00000  Median  : 12.118  Median  :4
## Mean    : 3.828  Mean   :0.08585  Mean   : 49.049  Mean   :4
## 3rd Qu.: 5.000  3rd Qu.:0.20000  3rd Qu.: 39.660  3rd Qu.:4
## Max.    :14.000  Max.   :0.80000  Max.   : 8399.976  Max.   :4
##

```

The first cluster has the majority values: ship mode is Second Class , segment is Consumer , City is San Francisco/other, State is California/other, sub category is furnishings. The mean difference in days between order date and ship date is 2.358 days, the mean value for Sales is \$216.38, the mean quantity is 3.699, the discount mean is \$0.1113, profit mean is \$32.

cluster 1 is made of second class shippng, Consumer segment x San Franciso x California x furnishings.

For cluster 2, the ship mode is mostly standard class, segment is mostly consumer, city is mostly Los Angeles/other, State is mostly California/Other, subcategory is mostly Binders/other. The mean difference in days between order date and ship date is 4.7 days, the mean sales \$225.91, mean quantity is 3.48, discount is 0.2148, and profit mean is \$19.67.

Cluster 2 is made of standard class shipping x consumer segment x Los Angeles x California x Binders.

For cluster 3, the main ship mode is First class, the segment is mostly Home Office, the most common city is Philadelphia/Other, the most common State is Pennsylvania/Other, and the most common subcategory is paper/other. For numeric variables, the mean difference in days between order date and ship date is 4.04 days, the mean sales \$216.94, mean quantity is 3.72, mean discount is 0.2065, and mean profit is \$10.08.

Cluster 3 is First class shipping x Home Office segment x Philadelphia x Pennsylvania x Paper. Cluster 3 has the lowest mean profit of the 4 clusters.

For cluster 4, the main ship mode is First class, the most common segment is Corporate, New York City/Other is the most common city, New York State/Other is the most common state, and the most common subcategory is paper. For the numeric features, the mean difference in days between order date and ship date is 4.25 days, the mean sales is \$253.84, and mean quantity is 3.83, mean discount is 0.086, mean profit is \$49.05.

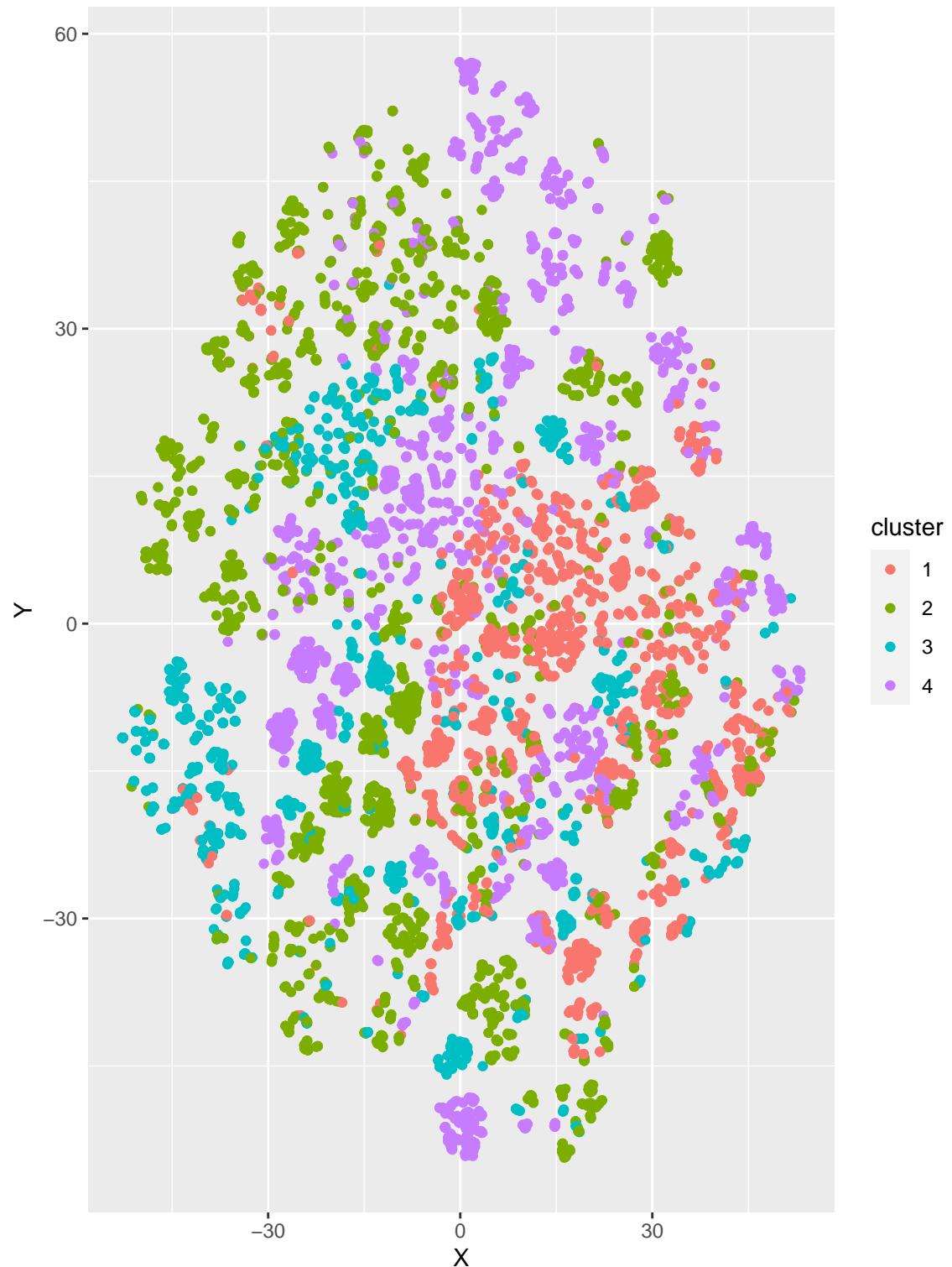
Cluster 4 is made of First Class shipping x Corporate segment x New York City x New York State x Paper. Cluster 4 has the highest mean sales amount and mean profit, with lowest mean discount.

We can now visualise the clusters in lower dimensional space with tSNE (t-Distributed Stochastic Neighbor Embedding) which can be used for dimensionality reduction.

```
tsne_obj <- Rtsne(gower_dist, is_distance = TRUE)

tsne_data <- tsne_obj$Y %>%
  data.frame() %>%
  setNames(c("X", "Y")) %>%
  mutate(cluster = factor(pam_fit$clustering))

ggplot(aes(x = X, y = Y), data = tsne_data) +
  geom_point(aes(color = cluster))
```



```
input_data <- data2
```

Let's try using k=3:

```

gower_dist = daisy(input_data, metric = "gower", type = list(logratio = 3))
gower_mat = as.matrix(gower_dist)
pam_fit = pam(gower_mat, k=3, diss=TRUE)

#summary(pam_fit)

```

to prevent printing out too many pages below, only some information from the summary of pam fit for k=3 will be included:

Medoids: ID

```
[1,] "399" "399" [2,] "9" "9"
[3,] "7119" "7119"
```

Numerical information per cluster: size max_diss av_diss diameter separation
[1,] 2361 0.6204122 0.3754422
0.7803200 0.01490751 [2,] 4281 0.6045552 0.3173436 0.8092413 0.01524512 [3,] 3352 0.5882532 0.3311972
0.8127438 0.01490751

Isolated clusters: L-clusters: character(0) L*-clusters: character(0)

Average silhouette width per cluster: [1] 0.06840880 0.09434220 0.06023282 Average silhouette width of total data set: [1] 0.07677532

Available components: [1] "medoids" "id.med" "clustering" "objective" "isolation" "clusinfo" "silinfo" "diss" "call"

Evaluating the consistency within Clusters of Data

Silhouette coefficient can be used to compare the average distance to observations within the same cluster, to the average distance to observations in other clusters.

High silhouette coefficient means that the observation is well clustered, while a low silhouette coefficient may indicate outliers.

Deployment

Responsible ML Framework

Conclusion

References

<https://www.datanovia.com/en/blog/k-means-clustering-visualization-in-r-step-by-step-guide/>
<https://www.tidymodels.org/learn/statistics/k-means/>
<https://towardsdatascience.com/clustering-analysis-in-r-using-k-means-73eca4fb7967>
<https://towardsdatascience.com/clustering-evaluation-strategies-98a4006fcfc>
<https://www.analyticsvidhya.com/blog/2020/12/a-case-study-to-detect-anomalies-in-time-series-using-anomaly-package-in-r/>
<https://www.datanovia.com/en/lessons/k-medoids-in-r-algorithm-and-practical-examples/>