Student ID: 0802653
Student Name: Tejalben Parmar
Subject: DAB 303 - Marketing Analytics
Instructor Name: Sarama Shehmir

Product Recommendation System

PROJECT MILESTONE 1

Problem Statement

Understanding marketing analytics enables companies or businesses to avoid missing out on their chance to show targeted recommendations based on user's preferences.

For the problem at hand, I will work with Walmart store transactions for online shopping. The objective is to analyze the data to find the insights and learn the customers' behaviors then segment them into groups to effectively target them individually involving new marketing strategies to achieve better outcomes.

Choice of Model / Statistical Methods

The first step is to perform EDA and then to find product and customer trend analysis to gain insights.

Next step would be applying Cohort Analysis and RFM Modeling, to divide customers into specific clusters based on their purchase histories.

Data Collection

Data to use for the project is downloaded from here

Dataset is in csv format named "SuperStoreOrders.csv" and it consists of following columns

category (string)
city (string)
container(string)
continent (string)
country_region (string)
customer_id (integer)
customer_name (string)
customer_segment (string)

department (string)
item (string)
order_date (date)
order_id (integer)
order_priority (string)
postal_code (string)
region (string)
row_id (integer)
ship_date (date)
ship_mode (string)
state (string)
discount (decimal)
number_of_records (boolean)
order_quantity (integer)
product_base_margin (decimal)
profit (integer)
sales (integer)
shipping_cost (integer)
unit_price (integer)

PROJECT MILESTONE 2

Importing required libraries

```
library(tidyverse)

## ── Attaching packages ──────────────────────────────────────────────
tidyverse 1.3.1 ──

## ✔ ggplot2 3.4.0      ✔ purrr   0.3.4
## ✔ tibble  3.1.7      ✔ dplyr   1.0.10
## ✔ tidyr   1.2.0      ✔ stringr 1.4.1
## ✔ readr   2.1.2      ✔ forcats 0.5.2

## Warning: package 'ggplot2' was built under R version 4.2.2

## Warning: package 'dplyr' was built under R version 4.2.1

## Warning: package 'stringr' was built under R version 4.2.1

## Warning: package 'forcats' was built under R version 4.2.2

## ── Conflicts ──────────────────────────────────────────────────────
tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
```

```
library("repr")

## Warning: package 'repr' was built under R version 4.2.2

library(stats)
library("dplyr")
library("ggplot2")
library("scales")

## Warning: package 'scales' was built under R version 4.2.2

##
## Attaching package: 'scales'

## The following object is masked from 'package:purrr':
##
##     discard

## The following object is masked from 'package:readr':
##
##     col_factor

library("lubridate")

## Warning: package 'lubridate' was built under R version 4.2.2

##
## Attaching package: 'lubridate'

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

library("ggcorrplot")

## Warning: package 'ggcorrplot' was built under R version 4.2.2

library("cohorts")

## Warning: package 'cohorts' was built under R version 4.2.2
```

Loading dataset from csv file and summarizing the dataset

```
df <-
read.csv("C:/Users/19054/Documents/Sem-3/303/Project/SuperStoreOrders.csv")
summary(df)

##    Category             City             Container          Continent
##  Length:16798       Length:16798       Length:16798       Length:16798
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
```

```
##
##  Country_Region      Customer_Id   Customer_Name      Customer_Segment
##  Length:16798       Min.   :   1   Length:16798       Length:16798
##  Class :character   1st Qu.: 912   Class :character   Class :character
##  Mode  :character   Median :1778   Mode  :character   Mode  :character
##                     Mean   :1754
##                     3rd Qu.:2593
##                     Max.   :3403
##
##   Department           Item          Order_Date          Order_Id
##  Length:16798       Length:16798       Length:16798       Min.   :    3
##  Class :character   Class :character   Class :character   1st Qu.:29858
##  Mode  :character   Mode  :character   Mode  :character   Median :72896
##                                                           Mean   :59335
##                                                           3rd Qu.:88699
##                                                           Max.   :91591
##
##  Order_Priority     Postal_Code          Region           Row_Id
##  Length:16798       Length:16798       Length:16798       Min.   :    1
##  Class :character   Class :character   Class :character   1st Qu.: 4200
##  Mode  :character   Mode  :character   Mode  :character   Median : 8400
##                                                           Mean   : 8400
##                                                           3rd Qu.:12599
##                                                           Max.   :16798
##
##   Ship_Date          Ship_Mode          State            Discount
##  Length:16798       Length:16798       Length:16798       Min.   :0.00000
##  Class :character   Class :character   Class :character   1st Qu.:0.02000
##  Mode  :character   Mode  :character   Mode  :character   Median :0.05000
##                                                           Mean   :0.04967
##                                                           3rd Qu.:0.08000
##                                                           Max.   :0.25000
##
##  Number_of_Records Order_Quantity   Product_Base_Margin     Profit
##  Min.   :1         Min.   :  1.00   Min.   :0.3500     Min.   :-17686.0
##  1st Qu.:1         1st Qu.:  8.00   1st Qu.:0.3800     1st Qu.:   -64.0
##  Median :1         Median : 16.00   Median :0.5200     Median :    12.0
##  Mean   :1         Mean   : 26.06   Mean   :0.5125     Mean   :   399.9
##  3rd Qu.:1         3rd Qu.: 38.00   3rd Qu.:0.5900     3rd Qu.:   229.0
##  Max.   :1         Max.   :180.00   Max.   :0.8500     Max.   : 60844.0
##                                     NA's   :126
##      Sales        Shipping_Cost      Unit_Price
##  Min.   :     1   Min.   :  0.00   Min.   :   1.00
##  1st Qu.:   100   1st Qu.:  3.00   1st Qu.:   6.00
##  Median :   360   Median :  6.00   Median :  21.00
##  Mean   :  1812   Mean   : 12.86   Mean   :  89.33
##  3rd Qu.:  1439   3rd Qu.: 14.00   3rd Qu.:  86.00
##  Max.   :100119   Max.   :165.00   Max.   :6783.00
##
```

Selecting project natives from the dataset for the project and have a look at the dataset

```
ProjectNatives <- c("Continent", "Country_Region", "Region", "State", "City",
"Customer_Segment", "Department", "Category", "Customer_Id", "Customer_Name",
"Order_Id", "Order_Date", "Order_Priority", "Item", "Container", "Ship_Date",
"Ship_Mode", "Discount", "Order_Quantity", "Profit", "Sales",
"Shipping_Cost", "Unit_Price")

store_data <- df[ProjectNatives]

head(store_data)

##       Continent           Country_Region  Region     State         City
## 1 North America United States of America Central  Michigan East Lansing
## 2 North America United States of America Central   Indiana       Carmel
## 3 North America United States of America Central Minnesota   Burnsville
## 4 North America United States of America Central  Missouri   Wentzville
## 5 North America United States of America Central   Indiana Merrillville
## 6 North America United States of America Central Minnesota      Hopkins
##   Customer_Segment Department               Category Customer_Id
## 1         Consumer  Furniture              Bookcases        1976
## 2         Consumer  Furniture                 Tables         596
## 3         Consumer  Furniture                 Tables        2204
## 4         Consumer  Furniture                 Tables        1789
## 5         Consumer  Furniture Chairs  and  Chairmats        1464
## 6         Consumer  Furniture Chairs  and  Chairmats        1522
##        Customer_Name Order_Id Order_Date Order_Priority
## 1    Sherri F Vogel     89039 2010-01-10       Critical
## 2 Doris Fitzpatrick     86308 2010-02-15       Critical
## 3        Oscar Ford     86053 2010-08-10       Critical
## 4       Allan Green     88261 2011-12-24       Critical
## 5   Evelyn Galloway     86398 2011-02-12       Critical
## 6        Earl Watts     89957 2010-12-14       Critical
##                                                              Item
Container
## 1                                     Hon Metal Bookcases, Putty  Jumbo
Box
## 2 Bretford Just In Time Height-Adjustable Multi-Task Work Tables  Jumbo
Box
## 3                                  Hon 94000 Series Round Tables  Jumbo
Box
## 4                                          BPI Conference Tables  Jumbo
Box
## 5                                         Hon GuestStacker Chair Jumbo
Drum
## 6                      Global High-Back Leather Tilter, Burgundy Jumbo
Drum
##    Ship_Date      Ship_Mode Discount Order_Quantity Profit Sales
Shipping_Cost
## 1 2010-01-11 Delivery Truck     0.05              8   -851   552
47
```

```
## 2 2010-02-16 Delivery Truck     0.07             12   -575  4911
75
## 3 2010-08-11 Delivery Truck     0.04             20    -88  5768
154
## 4 2011-12-25 Delivery Truck     0.03              6   -334   896
80
## 5 2011-02-14 Delivery Truck     0.03              6    934  1353
28
## 6 2010-12-15 Delivery Truck     0.10             17   -900  2027
70
##   Unit_Price
## 1         71
## 2        417
## 3        296
## 4        146
## 5        227
## 6        123
```

Looking at structure of the dataset

```
str(store_data)
```

```
## 'data.frame':    16798 obs. of  23 variables:
##  $ Continent       : chr  "North America" "North America" "North America"
"North America" ...
##  $ Country_Region  : chr  "United States of America" "United States of
America" "United States of America" "United States of America" ...
##  $ Region          : chr  "Central" "Central" "Central" "Central" ...
##  $ State           : chr  "Michigan" "Indiana" "Minnesota" "Missouri" ...
##  $ City            : chr  "East Lansing" "Carmel" "Burnsville"
"Wentzville" ...
##  $ Customer_Segment: chr  "Consumer" "Consumer" "Consumer" "Consumer" ...
##  $ Department      : chr  "Furniture" "Furniture" "Furniture" "Furniture"
...
##  $ Category        : chr  "Bookcases" "Tables" "Tables" "Tables" ...
##  $ Customer_Id     : int  1976 596 2204 1789 1464 1522 890 3228 2335 2447
...
##  $ Customer_Name   : chr  "Sherri F Vogel" "Doris Fitzpatrick" "Oscar
Ford" "Allan Green" ...
##  $ Order_Id        : int  89039 86308 86053 88261 86398 89957 89549 87439
89615 87791 ...
##  $ Order_Date      : chr  "2010-01-10" "2010-02-15" "2010-08-10"
"2011-12-24" ...
##  $ Order_Priority  : chr  "Critical" "Critical" "Critical" "Critical" ...
##  $ Item            : chr  "Hon Metal Bookcases, Putty" "Bretford Just In
Time Height-Adjustable Multi-Task Work Tables" "Hon 94000 Series Round
Tables" "BPI Conference Tables" ...
##  $ Container       : chr  "Jumbo Box" "Jumbo Box" "Jumbo Box" "Jumbo Box"
...
##  $ Ship_Date       : chr  "2010-01-11" "2010-02-16" "2010-08-11"
"2011-12-25" ...
```

```
##  $ Ship_Mode       : chr  "Delivery Truck" "Delivery Truck" "Delivery
Truck" "Delivery Truck" ...
##  $ Discount        : num  0.05 0.07 0.04 0.03 0.03 0.1 0.06 0.01 0.03 0.05
...
##  $ Order_Quantity  : int  8 12 20 6 6 17 8 11 1 1 ...
##  $ Profit          : int  -851 -575 -88 -334 934 -900 -1685 3764 -181 -215
...
##  $ Sales           : int  552 4911 5768 896 1353 2027 180 5456 125 174 ...
##  $ Shipping_Cost   : int  47 75 154 80 28 70 53 126 45 60 ...
##  $ Unit_Price      : int  71 417 296 146 227 123 21 501 101 159 ...

print(paste0("Total number of records in the dataset: ", nrow(store_data)))

## [1] "Total number of records in the dataset: 16798"
```

Removing NA records

```
store_data <- na.omit(store_data)
print(paste0("After removing NAs ", nrow(store_data), " records left"))

## [1] "After removing NAs 16798 records left"
```

Unique product items in the dataset

```
#unique(store_data$Item)
```

Removing item names containing only digits

```
# str_detect("e213", "^[:digit:]+$")
store_data <- store_data %>%
  filter(!(str_detect(store_data$Item, pattern = "^[:digit:]+$")))

print(paste0("Total number of records left in the dataset: ",
nrow(store_data)))

## [1] "Total number of records left in the dataset: 16416"
```

Extracting first word from the product name to populate Brand as a new column

```
store_data <- store_data %>%
  mutate(Brand = str_extract(store_data$Item, "(\\w+)"))

unique(store_data$Brand)

##   [1] "Hon"          "Bretford"      "BPI"          "Global"
##   [5] "Sauder"       "Iceberg"       "Office"       "Novimex"
##   [9] "Chromcraft"   "Westinghouse"  "OSullivan"    "Bush"
##  [13] "Bevis"        "Barricks"      "Riverside"    "SAFCO"
##  [17] "Atlantic"     "BoxOffice"     "DMI"          "Metal"
##  [21] "Rush"         "KI"            "Anderson"     "Safco"
##  [25] "Balt"         "Situations"    "Dana"         "Linden"
##  [29] "DAX"          "Master"        "Luxo"         "Magna"
##  [33] "Eldon"        "Tenex"         "Executive"    "Deflect"
##  [37] "Lesro"        "Howard"        "Seth"         "Lifetime"
##  [41] "G"            "Aluminum"      "Stacking"     "Staples"
##  [45] "Document"      "Laminate"      "6"             "Coloredge"
##  [49] "Nu"            "GE"            "Advantus"      "Regeneration"
##  [53] "12"            "3M"            "Career"        "Electrix"
##  [57] "Tensor"        "Telescoping"   "9"             "Flat"
##  [61] "Rubbermaid"    "Artistic"      "36X48"         "Ultra"
##  [65] "C"             "Hand"          "Contemporary"  "Computer"
##  [69] "1"             "Tennsco"       "Holmes"        "Avanti"
##  [73] "3"             "Sanyo"         "GBC"           "Xerox"
##  [77] "Project"       "Newell"        "Euro"          "Durable"
##  [81] "Eaton"         "Hot"           "File"          "White"
##  [85] "Letter"        "Fellowes"      "Black"         "Hunt"
##  [89] "Blue"          "Avery"         "Kensington"    "Trimflex"
##  [93] "Binder"        "Adams"         "Dixon"         "Wirebound"
##  [97] "Sanford"       "Belkin"        "Conquest"      "Cardinal"
## [101] "Crate"         "Harmony"       "Ames"          "Boston"
## [105] "Eureka"        "10"            "Important"     "Array"
## [109] "Premium"       "Space"         "Hoover"        "Catalog"
## [113] "Peel"          "Prang"         "Panasonic"     "Wilson"
## [117] "Acco"          "Heavy"         "Vinyl"         "Colored"
## [121] "Multimedia"    "Acme"          "Fiskars"       "Snap"
## [125] "While"         "DIXON"         "Tripp"         "Brites"
## [129] "Stockwell"     "Honeywell"     "Iris"          "Storex"
## [133] "Southworth"    "Ibico"         "Barrel"        "Rediform"
## [137] "Plymouth"      "Economy"       "SANFORD"       "Park"
## [141] "Fluorescent"   "Sterling"      "Telephone"     "Unpadded"
## [145] "Quartet"       "Decoflex"      "Lock"          "Crayola"
## [149] "2300"          "XtraLife"      "HP"            "Gould"
## [153] "Filing"        "Bagged"        "Portfile"      "Jet"
## [157] "Surelock"      "Recycled"      "Portable"      "Prismacolor"
## [161] "Its"           "Binney"        "BOSTON"        "Home"
## [165] "REDIFORM"      "Serrated"      "Mead"          "Angle"
## [169] "Astroparche"   "Ampad"         "Martin"        "OIC"
## [173] "Alliance"      "TOPS"          "EcoTones"      "Multicolor"
## [177] "Dot"           "24"            "Turquoise"     "ACCOHIDE"
## [181] "Super"         "Speediset"     "Berol"         "Manila"
## [185] "Carina"        "Binding"       "Large"         "Pressboard"
## [189] "Memo"          "Spiral"        "Avoid"         "Presstex"
## [193] "Bionaire"      "Desktop"       "Revere"        "JM"
## [197] "Dual"          "Kleencut"      "Self"          "Wausau"
## [201] "Quality"       "DXL"           "Perma"         "Trav"
## [205] "Assorted"      "Poly"          "Smead"         "Deluxe"
## [209] "Steel"         "Sensible"      "Premier"       "Multi"
## [213] "Rogers"        "Riverleaf"     "Personal"      "SimpliFile"
## [217] "Lumber"        "Message"       "4009"          "Bravo"
## [221] "Tyvek"         "Tuff"          "Sterilite"     "Zebra"
## [225] "Companion"     "Strathmore"    "Standard"      "Hanging"
## [229] "X"             "Security"      "Universal"     "Flexible"
## [233] "Airmail"       "IBM"           "Elite"         "Hammermill"
## [237] "Accohide"      "APC"           "Brown"         "Laser"
## [241] "Round"         "Model"         "High"          "Satellite"
## [245] "Pizazz"        "Grip"          "Plastic"       "Rubber"
## [249] "Stanley"       "Post"          "Geographics"   "Blackstonian"
## [253] "Col"           "UniKeep"       "14"            "Colorific"
## [257] "Hewlett"       "Lexmark"       "Okidata"       "Epson"
## [261] "Canon"         "Sharp"         "Adesso"        "CF"
## [265] "U"             "StarTAC"       "Accessory4"    "KH"
## [269] "Logitech"      "2160i"         "KF"            "Verbatim"
## [273] "Accessory21"   "600"           "Imation"       "Accessory6"
## [277] "Accessory2"    "Targus"        "Talkabout"     "i270"
## [281] "Zoom"          "Accessory12"   "270c"          "Bell"
## [285] "R380"          "Gyration"      "Polycom"       "Micro"
## [289] "Memorex"       "Soundgear"     "Accessory8"    "Motorola"
## [293] "80"            "PC"            "M3682"         "MicroTAC"
## [297] "DS"            "V"             "Accessory27"   "Timeport"
## [301] "Accessory41"   "TDK"           "Microsoft"     "Hayes"
## [305] "T39m"          "6162m"         "Accessory32"   "Accessory9"
## [309] "Accessory28"   "i500plus"      "US"            "g520"
## [313] "Phone"         "AT"            "T65"           "Brother"
## [317] "1726"          "Keytronic"     "Accessory36"   "Maxell"
## [321] "V70"           "Accessory39"   "T60"           "DPC"
## [325] "Accessory37"   "i1000"         "LX"            "TimeportP7382"
## [329] "i470"          "Accessory34"   "V2397"         "300"
## [333] "VTech"         "Accessory35"   "5170i"         "T193"
## [337] "i600"          "TI"            "T28"           "Fuji"
## [341] "Accessory15"   "SouthWestern"  "Accessory17"   "Accessory20"
## [345] "Accessory13"   "BASF"          "Sony"          "SC"
## [349] "iDEN"          "Accessory29"   "i2000"         "Accessory25"
## [353] "Accessory31"   "i1000plus"     "ELITE"         "210"
## [357] "SC7868i"       "6162i"         "TIMEPORT"      "T18"
## [361] "R280"          "A1228"         "I888"          "M70"
## [365] "iDENi80s"      "T61"           "Accessory24"   "V3682"
## [369] "V8162"         "V8160"         "R289LX"        "Accessory23"
## [373] "Accessory1"    "V66"
```

Replacing brand names containing only digits with “Unknown”

```
#str_detect("213", "^[:digit:]+$")
store_data$Brand <- str_replace(store_data$Brand, "^[:digit:]+$", "Unknown")

#unique(store_data$Brand)

print(paste0("Total number of unique items in the dataset: ",
length(unique(store_data$Item))))

## [1] "Total number of unique items in the dataset: 1231"

print(paste0("Total number of unique brands in the dataset: ",
length(unique(store_data$Brand))))

## [1] "Total number of unique brands in the dataset: 360"
```

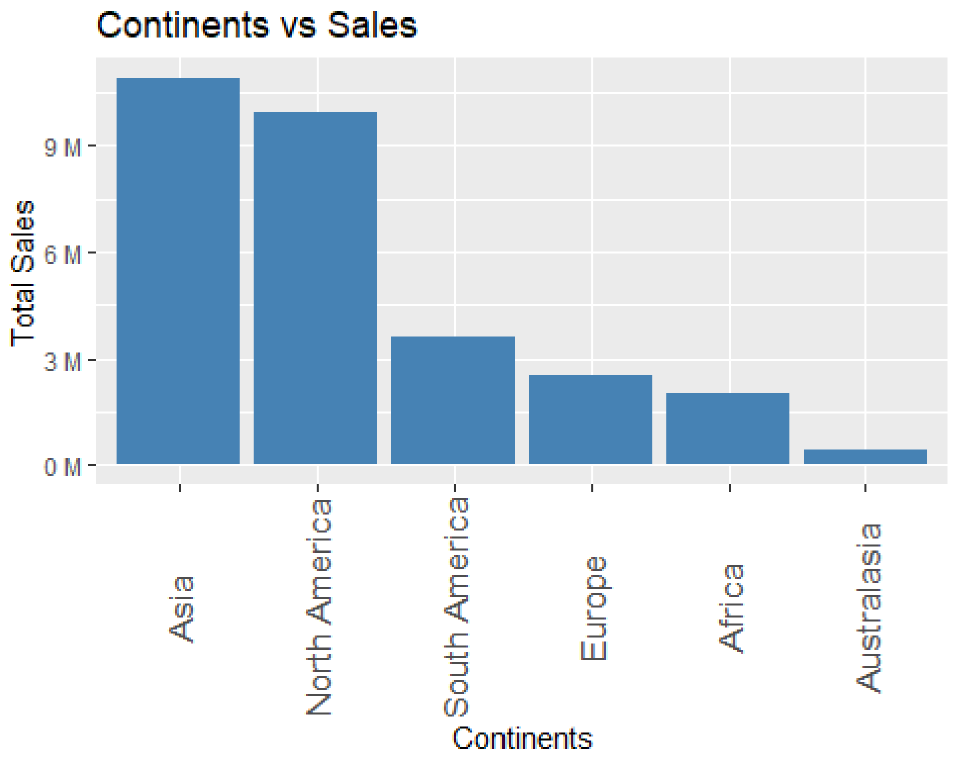
```
#
write.csv(store_data,"C:/Users/19054/Documents/Sem-3/303/Project/store_data_s
elected.csv", row.names = FALSE)
```

Sales comparision in different Continents

```
#Aggregating data by 'Continent' and Finding sum of 'Sales'
Continent_Sales<- aggregate(Sales ~ Continent, data = store_data, sum)

#Changing column name of sales
colnames(Continent_Sales)[2] <- "Total_Sales"

#Finding out Store with highest Sales

Continent_Sales <-arrange(Continent_Sales, desc(Total_Sales)) #Arranged
Continents based on Sales in descending order
Continent_Sales[]
```

```
##       Continent Total_Sales
## 1          Asia    10913895
## 2 North America     9914985
## 3 South America     3632053
## 4        Europe     2541709
## 5        Africa     2014823
## 6   Australasia      411434
```

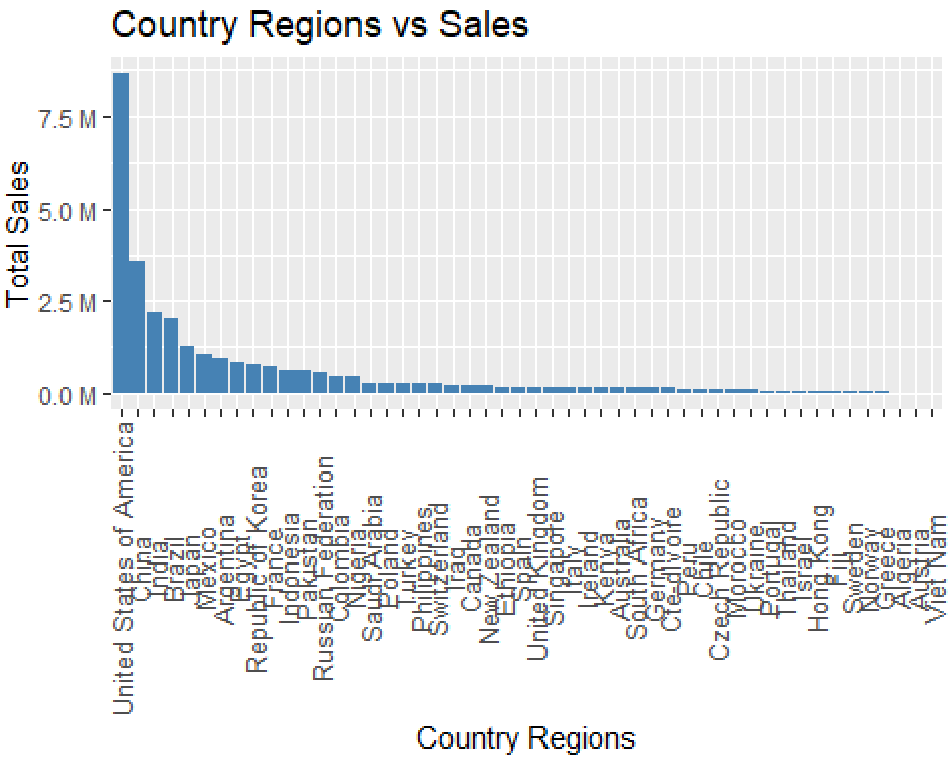
```
# Converting Continent column into factor so that order won't change for
graph
Continent_Sales$Continent <- factor(Continent_Sales$Continent, levels =
unique(Continent_Sales$Continent))

#Plotting Continent vs TotalSales

ggplot(data = Continent_Sales, aes(x = Continent, y = Total_Sales)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 0.5, size
= 13)) +
  scale_y_continuous(labels = label_number(suffix = " M", scale = 1e-6)) +
ggtitle('Continents vs Sales') +
  xlab("Continents") + ylab("Total Sales")
```



Sales comparision in different Countries

```
#Aggregating data by 'Country' and Finding sum of 'Sales'
Country_Sales <- aggregate(Sales ~ Country_Region, data = store_data, sum)

#Changing column name of sales
colnames(Country_Sales)[2] <- "Total_Sales"

#Finding out Country with highest Sales
Country_Sales <-arrange(Country_Sales, desc(Total_Sales)) #Arranged
Continents based on Sales in descending order
Country_Sales[]

##                Country_Region Total_Sales
## 1  United States of America     8659432
## 2                     China     3546284
## 3                     India     2167687
## 4                    Brazil     2023342
## 5                     Japan     1273633
## 6                    Mexico     1053650
## 7                 Argentina      959590
## 8                     Egypt      804026
## 9         Republic of Korea      785651
## 10                   France      723043
## 11                Indonesia      591663
## 12                 Pakistan      581054
## 13       Russian Federation      547901
## 14                 Colombia      443995
## 15                  Nigeria      436100
## 16             Saudi Arabia      294808
## 17                   Poland      291104
## 18                   Turkey      281600
## 19              Philippines      257641
## 20              Switzerland      250140
## 21                     Iraq      212935
## 22                   Canada      201903
## 23              New Zealand      196150
## 24                 Ethiopia      194020
## 25                    Spain      187087
## 26           United Kingdom      186959
## 27                Singapore      175984
## 28                    Italy      169125
## 29                  Ireland      169019
## 30                    Kenya      165214
## 31                Australia      158689
## 32             South Africa      155035
## 33                  Germany      147581
## 34              Cte-dIvoire      141609
## 35                     Peru      105802
## 36                    Chile       99324
## 37           Czech Republic       98735
## 38                  Morocco       92351
## 39                  Ukraine       85997
## 40                 Portugal       81393
## 41                 Thailand       68058
## 42                   Israel       59846
## 43                Hong Kong       58605
## 44                     Fiji       56595
## 45                   Sweden       49088
## 46                   Norway       43415
## 47                   Greece       42782
## 48                  Algeria       26468
## 49                  Austria       16241
## 50                 Viet Nam       10545
```

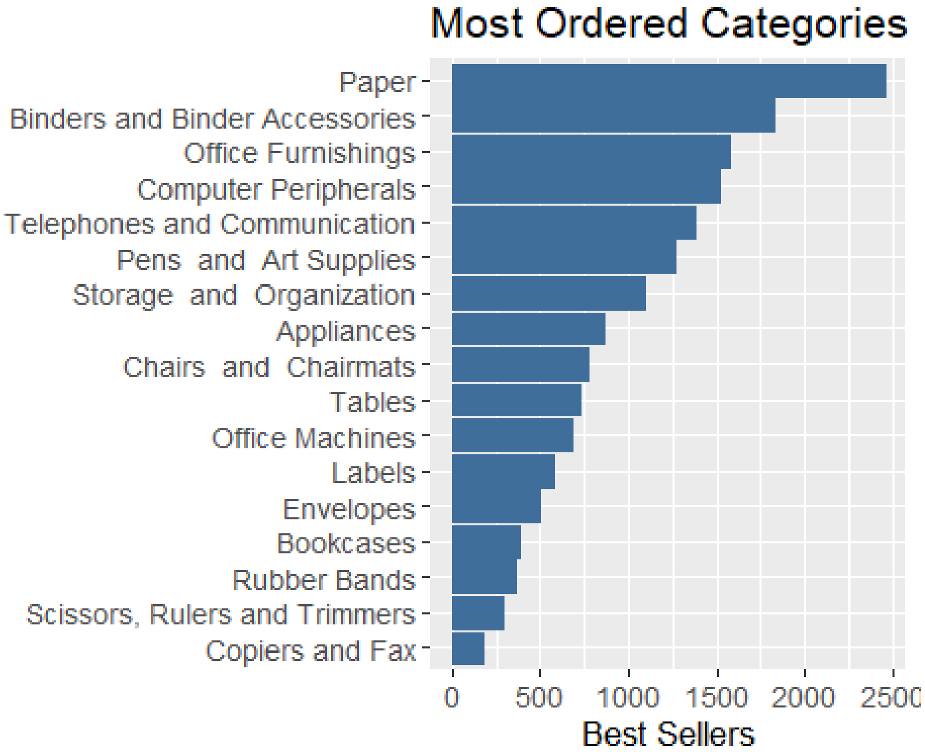
```
# Converting Country_Region column into factor so that order won't change for
graph
Country_Sales$Country_Region <- factor(Country_Sales$Country_Region, levels =
unique(Country_Sales$Country_Region))

#Plotting Country_Region vs TotalSales

#options(repr.plot.width = 30, repr.plot.height = 20)

ggplot(data = Country_Sales, aes(x = Country_Region, y = Total_Sales)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 0.5, size
= 10)) +
  scale_y_continuous(labels = label_number(suffix = " M", scale = 1e-6)) +
ggtitle('Country Regions vs Sales') +
  xlab("Country Regions") + ylab("Total Sales")
```



Identifying Most Ordered Categories

```
categories <- store_data %>%
  group_by(Category) %>%
  summarize(count = n()) %>%
  arrange(desc(count))

categories

## # A tibble: 17 × 2
##    Category                        count
##    <chr>                           <int>
##  1 Paper                            2450
##  2 Binders and Binder Accessories   1830
##  3 Office Furnishings               1576
##  4 Computer Peripherals             1516
##  5 Telephones and Communication     1384
##  6 Pens  and  Art Supplies          1266
##  7 Storage  and  Organization       1092
##  8 Appliances                        868
##  9 Chairs  and  Chairmats            772
## 10 Tables                            722
## 11 Office Machines                    674
## 12 Labels                             576
## 13 Envelopes                          492
## 14 Bookcases                          378
## 15 Rubber Bands                       358
## 16 Scissors, Rulers and Trimmers      288
## 17 Copiers and Fax                    174
```

```
ggplot(data = categories, aes(x = reorder(Category, count), y = count))+
  geom_bar(stat = "identity", fill = "#3F6E9A", colour = "#3F6E9A") +
  labs(x = "", y = "Best Sellers", title = "Most Ordered Categories") +
  coord_flip() +
  theme(text = element_text(size = 13))
```

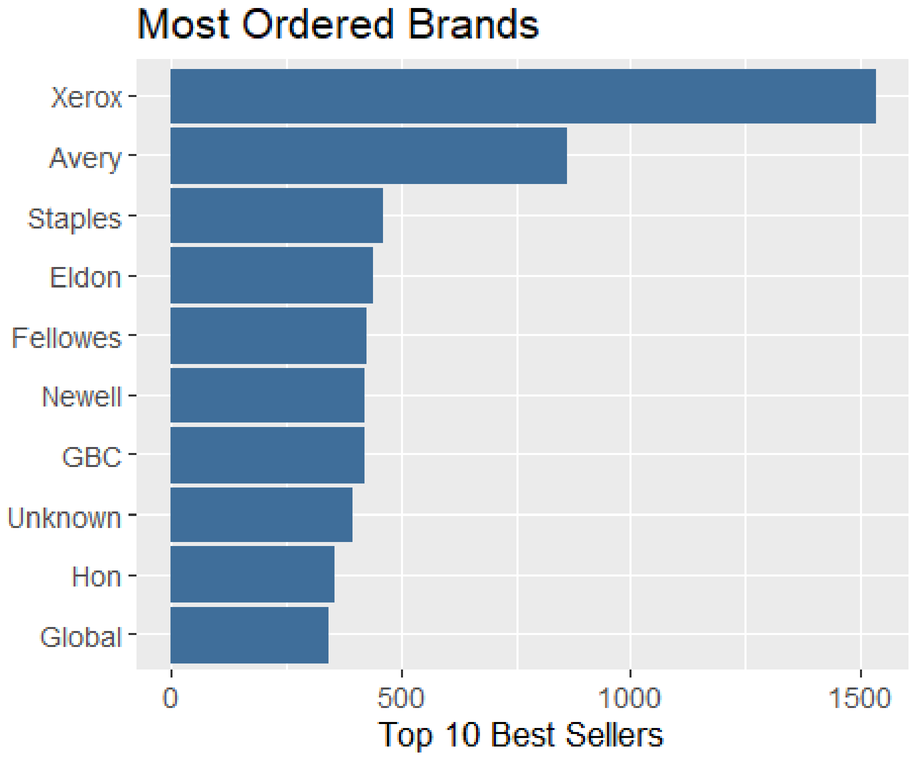


Identifying most ordered Brands

```
brands <- store_data %>%
  group_by(Brand) %>%
  summarize(count = n()) %>%
  arrange(desc(count))

brands
```

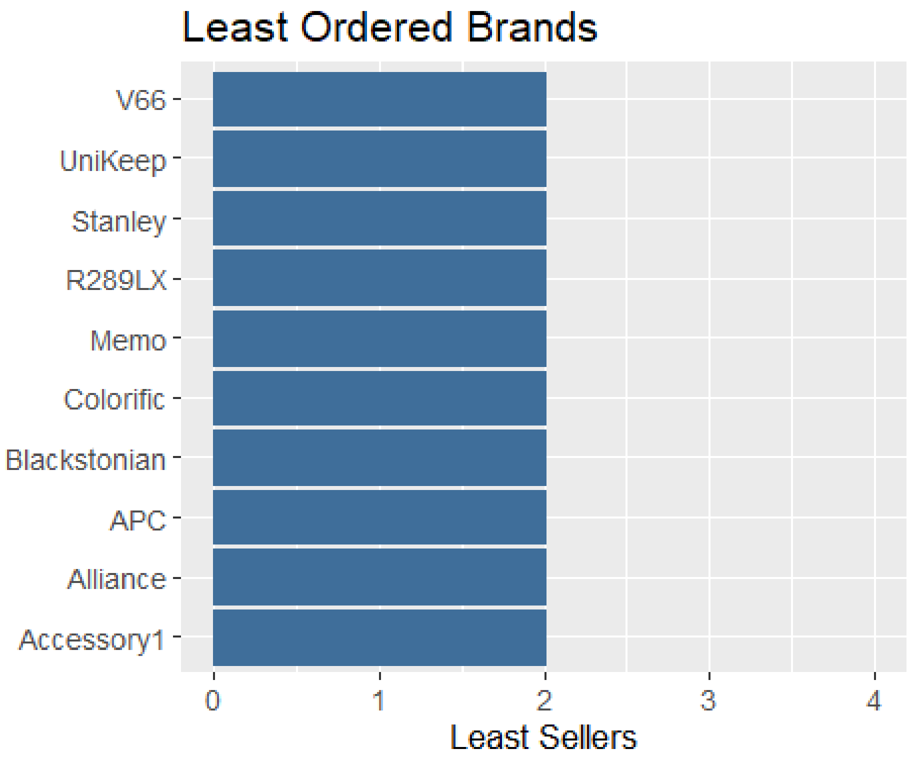
```
## # A tibble: 360 × 2
##    Brand    count
##    <chr>    <int>
##  1 Xerox     1530
##  2 Avery      858
##   3 Staples    458
##   4 Eldon      434
##   5 Fellowes   422
##   6 GBC        416
##   7 Newell     416
##   8 Unknown    388
##   9 Hon        352
##  10 Global     338
## # ... with 350 more rows
```

```
ggplot(data = brands[0:10, ], aes(x = reorder(Brand, count), y = count))+
  geom_bar(stat = "identity", fill = "#3F6E9A", colour = "#3F6E9A") +
  labs(x = "", y = "Top 10 Best Sellers", title = "Most Ordered Brands") +
  coord_flip() +
  theme(text = element_text(size = 13))
```



Identifying Least ordered Brands

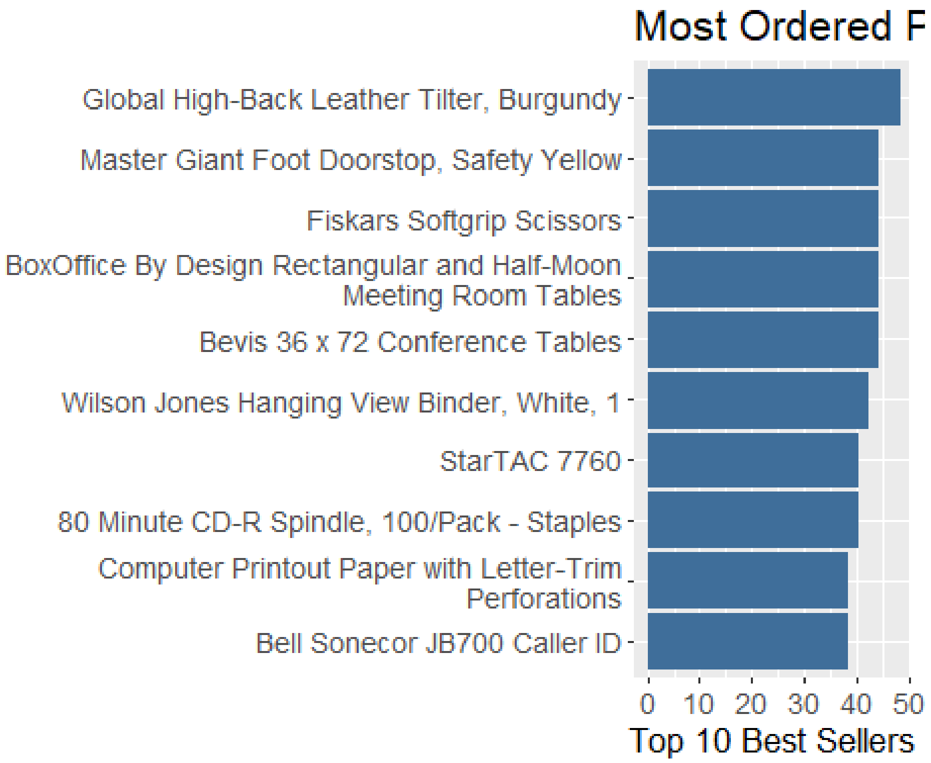
```
ggplot(data = tail(brands, n = 10), aes(x = reorder(Brand, -count), y =
count))+
  geom_bar(stat = "identity", fill = "#3F6E9A", colour = "#3F6E9A") +
  labs(x = "", y = "Least Sellers", title = "Least Ordered Brands") +
  scale_y_continuous(limits = c(0, 4), breaks = c(0, 1, 2, 3, 4)) +
  coord_flip() +
  theme(text = element_text(size = 13))
```



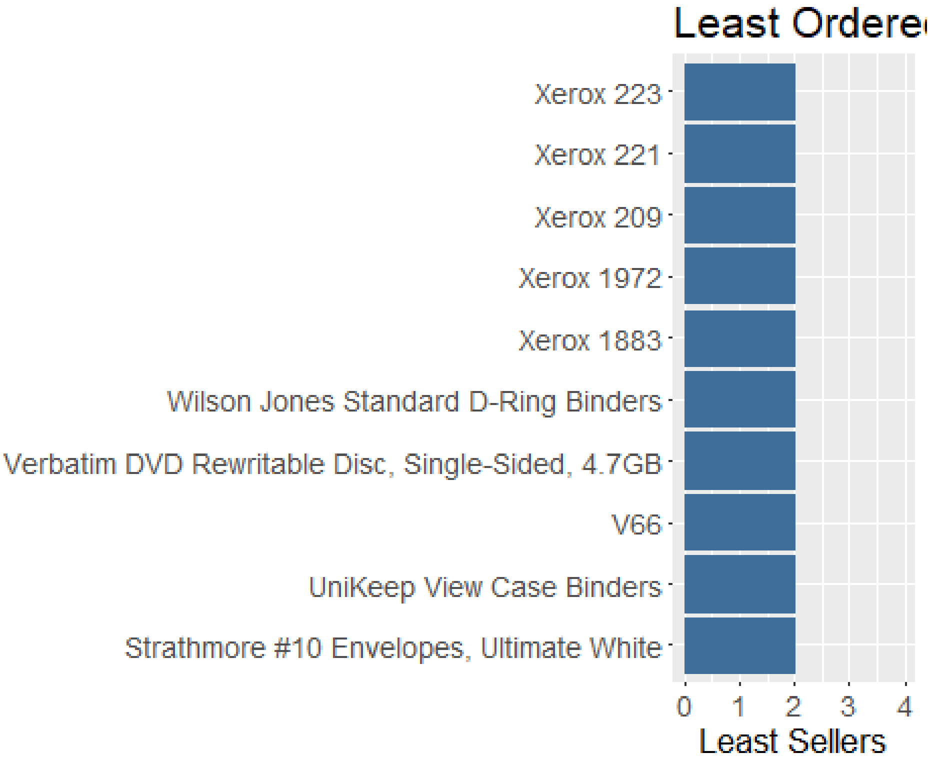
Identifying most ordered Products

```
# store_data_cleaned <- as.data.frame(gsub("[[:punct:]]", "",
as.matrix(store_data)))
products <- store_data %>%
  group_by(Item) %>%
  summarize(count = n()) %>%
  arrange(desc(count))

products

## # A tibble: 1,231 × 2
##    Item                                                        count
##    <chr>                                                       <int>
##  1 Global High-Back Leather Tilter, Burgundy                      48
##  2 Bevis 36 x 72 Conference Tables                                44
##  3 BoxOffice By Design Rectangular and Half-Moon Meeting Room Tables    44
##  4 Fiskars Softgrip Scissors                                      44
##  5 Master Giant Foot Doorstop, Safety Yellow                      44
##  6 Wilson Jones Hanging View Binder, White, 1                     42
##  7 80 Minute CD-R Spindle, 100/Pack - Staples                     40
##  8 StarTAC 7760                                                   40
##  9 Bell Sonecor JB700 Caller ID                                   38
## 10 Computer Printout Paper with Letter-Trim Perforations          38
## # … with 1,221 more rows
```

```
ggplot(data = products[0:10, ], aes(x = reorder(Item, count), y = count))+
  geom_bar(stat = "identity", fill = "#3F6E9A", colour = "#3F6E9A") +
  labs(x = "", y = "Top 10 Best Sellers", title = "Most Ordered Products") +
  coord_flip() +
  scale_x_discrete(labels = function(x) str_wrap(x, width = 50)) +
  theme_grey(base_size = 10) +
  theme(text = element_text(size = 13))
```

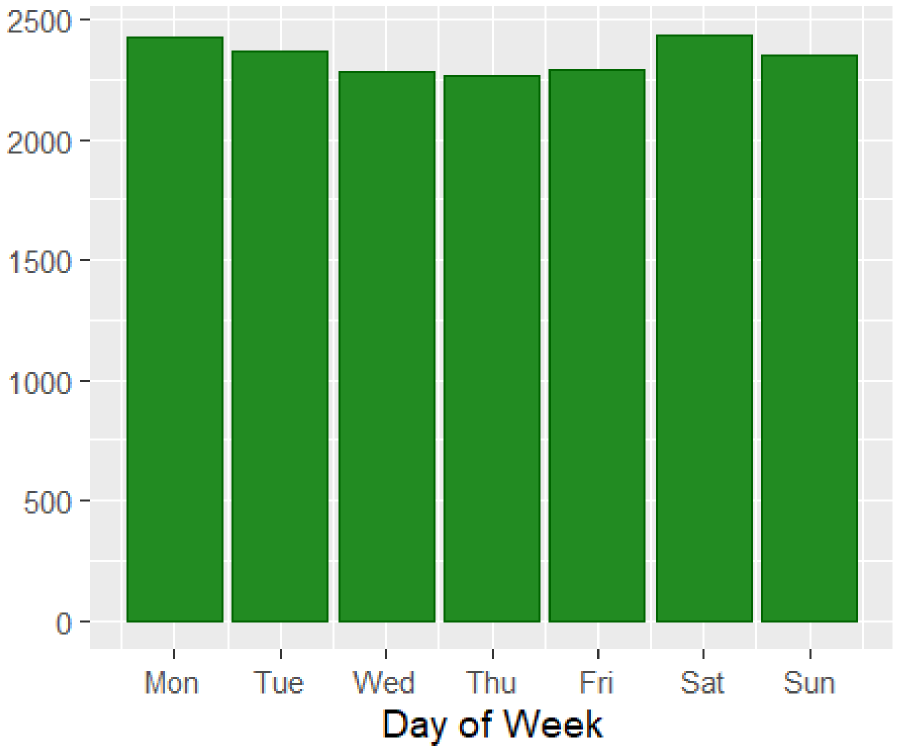


```
ggplot(data = tail(products, n = 10), aes(x = reorder(Item, -count), y =
count))+
  geom_bar(stat = "identity", fill = "#3F6E9A", colour = "#3F6E9A") +
  labs(x = "", y = "Least Sellers", title = "Least Ordered Products") +
  scale_y_continuous(limits = c(0, 4), breaks = c(0, 1, 2, 3, 4)) +
  scale_x_discrete(labels = function(x) str_wrap(x, width = 50)) +
  coord_flip() +
  theme_grey(base_size = 8) +
  theme(text = element_text(size = 13))
```



Frequency of orders on different week days

```
store_data %>%
  ggplot(aes(wday(Order_Date, week_start = getOption("lubridate.week.start",
1)))) +
  geom_histogram(stat = "count" , fill = "forest green", colour = "dark
green") +
  labs(x = "Day of Week", y = "") +
  scale_x_continuous(breaks = c(1,2,3,4,5,6,7), labels = c("Mon", "Tue",
"Wed", "Thu", "Fri", "Sat", "Sun")) +
  theme_grey(base_size = 14)

## Warning in geom_histogram(stat = "count", fill = "forest green", colour =
"dark
## green"): Ignoring unknown parameters: `binwidth`, `bins`, and `pad`
```



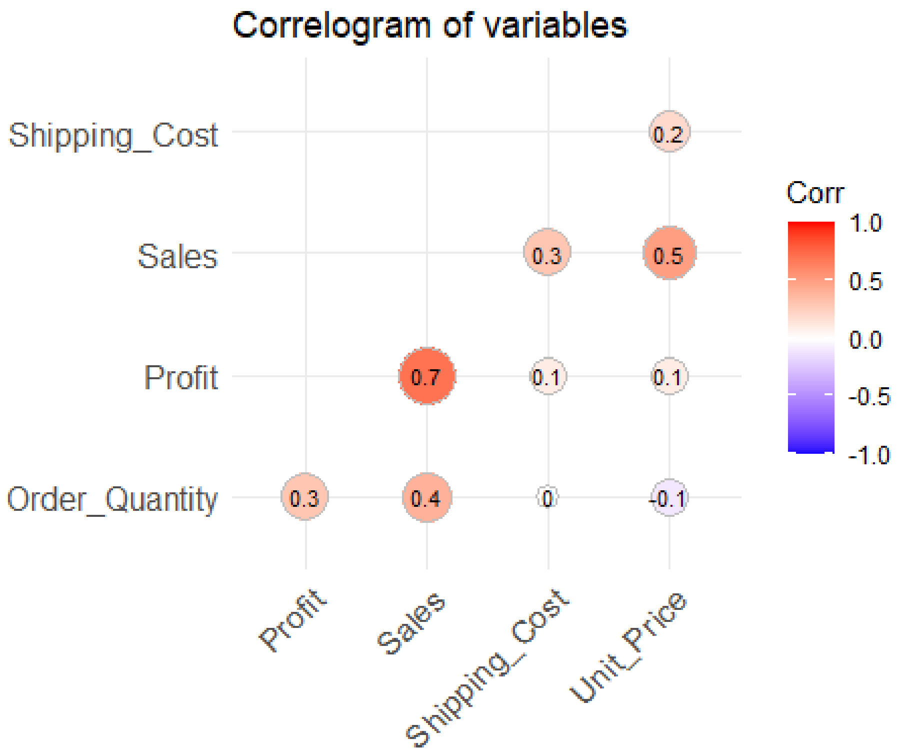
Relationships among numerical variables

```
cordata = store_data[,c(19, 20, 21, 22, 23)]
corr <- round(cor(cordata), 1)
corr
```

```
##                Order_Quantity Profit Sales Shipping_Cost Unit_Price
## Order_Quantity            1.0    0.3   0.4           0.0       -0.1
## Profit                    0.3    1.0   0.7           0.1        0.1
## Sales                     0.4    0.7   1.0           0.3        0.5
## Shipping_Cost             0.0    0.1   0.3           1.0        0.2
## Unit_Price               -0.1    0.1   0.5           0.2        1.0
```

The output above shows the presence of strong linear correlation between the variables Profit and Sales

```
ggcorrplot(corr, hc.order = TRUE, type = "lower", lab = TRUE, lab_size = 3,
method="circle", colors = c("blue", "white", "red"), outline.color = "gray",
show.legend = TRUE, show.diag = FALSE, title="Correlogram of variables")
```



Correlogram of variables
Shipping_Cost
Sales
Profit
Order_Quantity
0.2
0.3
0.5
0.7
0.1
0.1
0.3
0.4
0
-0.1
Profit
Sales
Shipping_Cost
Unit_Price
Corr
1.0
0.5
0.0
-0.5
-1.0