SUPERSTORE SALES ANALYSIS

### Data Source

For this project, I leveraged a dataset from Kaggle, specifically the “Superstore Sales Dataset” by Rohit Sahoo. This dataset contains information regarding a retail data for 4 years (2015-2018).

### Problem Statement

In the retail industry, understanding sales, customer behavior and product trends is essential for businesses striving to optimize their operations and enhance customer satisfaction. This dataset focus on online retail transactions, presents an opportunity to uncover critical aspects influencing business performance.

### Objective

The following are business related questions to our data:

1. What was the best month for sales? How much was earned that month?  
2. Top 10 states sold the most products?  
3. At what time do we have the highest sales?  
4. Are there certain times the number of items sold peaks or dip significantly?   
5. What top 10 product sold the most?

### Data Preparation

#Loading libraries  
library(janitor)  
library(lubridate)  
library(tidyverse)  
library(webr)  
library(ggplot2)  
library(scales)  
library(maps)

### Data Cleaning

#Changing column names   
Data <- clean\_names(Data)  
colnames(Data)

#Checking data types  
str(Data)  
#Changing data types for order\_date and ship\_date columns  
Data$order\_date <- as.Date(Data$order\_date, '%d/%m/%Y')  
Data$ship\_date <- as.Date(Data$ship\_date, '%d/%m/%Y')  
str(Data)

#Checking for null values  
sum(is.na(Data))  
#Where are the null values  
sapply(Data, function(x)which(is.na(x)))  
#Highlight the rows with null values  
null\_values <- Data[c(2235, 5275, 8799, 9147, 9148, 9149, 9387, 9388, 9389, 9390, 9742),]  
null\_values  
#The null values are the Postal Code for Vermont, Burlington, United States.  
#Replacing the null with the correct Postal Code of Burlington city in Vermont state ; 05401-05408  
Data$postal\_code <- replace\_na(Data$postal\_code,05401)

#Checking for duplicate rows  
sum(duplicated(Data))

#Attach the data  
attach(Data)

### Data Exploration

#### Order Analysis

#Total number of orders:  
length(order\_id)

## [1] 9800

#Total number of unique orders:  
length(unique(order\_id))

## [1] 4922

#Order sales  
sales\_by\_order\_id <- aggregate(sales~order\_id, data = Data, sum) %>%  
 arrange(desc(sales))  
#Top 5 orders by sum  
top\_orders <- head(sales\_by\_order\_id, 5)  
top\_orders

## order\_id sales  
## 1 CA-2015-145317 23661.23  
## 2 CA-2017-118689 18336.74  
## 3 CA-2018-140151 14052.48  
## 4 CA-2018-127180 13716.46  
## 5 CA-2015-139892 10539.90

#Bottom 5 orders by sum  
bottom\_orders <- tail(sales\_by\_order\_id, 5)  
bottom\_orders

## order\_id sales  
## 4918 US-2018-100209 1.080  
## 4919 US-2015-152723 0.876  
## 4920 CA-2015-112403 0.852  
## 4921 CA-2017-168361 0.836  
## 4922 CA-2018-124114 0.556

Insight:

1. A total of 9800 orders and 4922 unique orders were made from Jan 2015 to Dec 2018  
2. Order 'CA-2015-145317' leads in sales with $23661.23  
3. Order 'CA-2018-124114' has the lowest sales with $0.556.

#### Category and Sub-category Analysis

#Main category sales  
category\_sum <- aggregate(sales~category, data = Data, sum) %>%  
 arrange(desc(sales))  
category\_sum

## category sales  
## 1 Technology 827455.9  
## 2 Furniture 728658.6  
## 3 Office Supplies 705422.3

#Sub-category sales  
sub\_category\_sum <- aggregate(sales~sub\_category, data = Data, sum) %>%  
 arrange(desc(sales))  
sub\_category\_sum

## sub\_category sales  
## 1 Phones 327782.45  
## 2 Chairs 322822.73  
## 3 Storage 219343.39  
## 4 Tables 202810.63  
## 5 Binders 200028.79  
## 6 Machines 189238.63  
## 7 Accessories 164186.70  
## 8 Copiers 146248.09  
## 9 Bookcases 113813.20  
## 10 Appliances 104618.40  
## 11 Furnishings 89212.02  
## 12 Paper 76828.30  
## 13 Supplies 46420.31  
## 14 Art 26705.41  
## 15 Envelopes 16128.05  
## 16 Labels 12347.73  
## 17 Fasteners 3001.96

#Mean of sales by categories:  
round(sum(sales)/length(unique(category)),2)

## [1] 753845.6

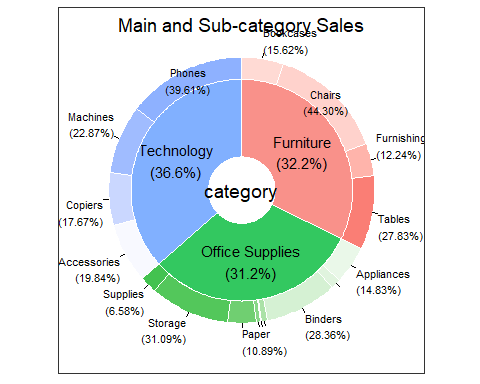
#Mean of sales by sub-categories:  
round(sum(sales)/length(unique(sub\_category)),2)

## [1] 133031.6

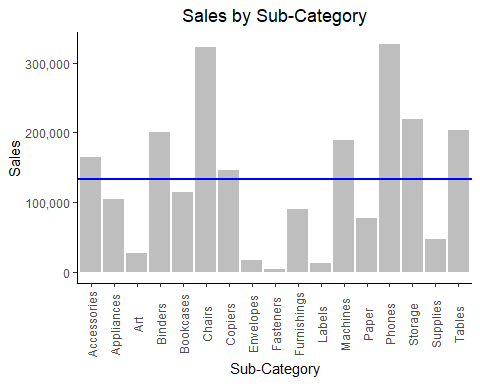
#Pie plot  
table\_1 <- Data %>%  
 group\_by(category, sub\_category) %>%  
 summarise(sales = sum(sales))

## `summarise()` has grouped output by 'category'. You can override using the  
## `.groups` argument.

PieDonut(table\_1,  
 aes(category, sub\_category, count = sales),  
 title = "Main and Sub-category Sales")



#Sub-category bar plot  
ggplot(data = Data, aes(x = sub\_category, y = sales)) +  
 geom\_bar(stat = "identity", fill = "grey") +  
 geom\_hline(yintercept = sum(sales)/length(unique(sub\_category)), color = "blue", lwd = 0.75) +  
 scale\_y\_continuous(labels = comma) +  
 labs(title = "Sales by Sub-Category", x = "Sub-Category", y = "Sales") +  
 theme\_classic() +  
 theme(plot.title = element\_text(hjust = 0.5), axis.text.x = element\_text(angle = 90, vjust = 0.5))



Insight:

1. In main category, "Technology" has the highest sales with 36.6% and Total Sales of $827455.9  
2. In sub-category, "Phones" has the highest sales with 39.61% and Total Sales of $327782.45  
3. Out of 17 only 8 sub-categories of products had Total Sales above the Average Sales.

##### Product Analysis

#Product rankings  
product\_ranking <- Data %>%  
 group\_by(product\_id) %>%  
 summarise(total\_orders = length(product\_id), average\_price = mean(sales)) %>%  
 arrange(desc(total\_orders))  
#Top 10 products  
top\_products <- head(product\_ranking, 10)  
top\_products

## # A tibble: 10 × 3  
## product\_id total\_orders average\_price  
## <chr> <int> <dbl>  
## 1 OFF-PA-10001970 19 114.   
## 2 TEC-AC-10003832 18 622.   
## 3 FUR-FU-10004270 16 26.7  
## 4 FUR-CH-10002647 15 243.   
## 5 TEC-AC-10002049 15 917.   
## 6 TEC-AC-10003628 15 94.0  
## 7 FUR-CH-10001146 14 161.   
## 8 FUR-CH-10002880 14 388.   
## 9 FUR-CH-10003774 14 298.   
## 10 FUR-FU-10001473 14 60.0

Insight:

The most ordered product is 'OFF-PA-10001970' with 19 orders and Average Price of $114.07

##### Date Analysis

#Total sales by year  
year\_sales <- Data %>%  
 group\_by(year = year(order\_date)) %>%  
 summarize(total\_sales = sum(sales)) %>%  
 mutate(rate = (total\_sales - dplyr::lag(total\_sales))/dplyr::lag(total\_sales)\*100)  
year\_sales

## # A tibble: 4 × 3  
## year total\_sales rate  
## <dbl> <dbl> <dbl>  
## 1 2015 479856. NA   
## 2 2016 459436. -4.26  
## 3 2017 600193. 30.6   
## 4 2018 722052. 20.3

#Average sales by month and total orders over the years  
month\_sales <- Data %>%  
 group\_by(month = month(order\_date)) %>%  
 summarize(total\_orders = length(order\_id), average\_sales = mean(sales)) %>%  
 mutate(average\_rate = (average\_sales - dplyr::lag(average\_sales))/dplyr::lag(average\_sales)\*100,   
 orders\_rate = (total\_orders - dplyr::lag(total\_orders))/dplyr::lag(total\_orders)\*100)  
month\_sales

## # A tibble: 12 × 5  
## month total\_orders average\_sales average\_rate orders\_rate  
## <dbl> <int> <dbl> <dbl> <dbl>  
## 1 1 366 258. NA NA   
## 2 2 297 200. -22.4 -18.9   
## 3 3 680 291. 45.3 129.   
## 4 4 657 207. -28.6 -3.38   
## 5 5 725 213. 2.46 10.4   
## 6 6 691 211. -0.697 -4.69   
## 7 7 697 209. -1.07 0.868  
## 8 8 693 227. 8.72 -0.574  
## 9 9 1354 222. -2.36 95.4   
## 10 10 809 247. 11.3 -40.3   
## 11 11 1449 242. -2.00 79.1   
## 12 12 1382 233. -3.74 -4.62

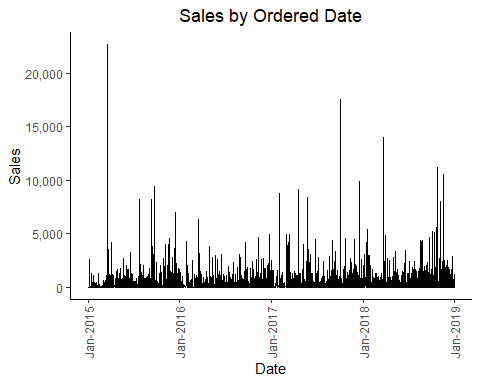
#Time with highest sales  
time\_sales <- Data %>%  
 group\_by(order\_date) %>%  
 summarize(total\_sales = sum(sales)) %>%  
 arrange(desc(total\_sales))  
head(time\_sales,3)

## # A tibble: 3 × 2  
## order\_date total\_sales  
## <date> <dbl>  
## 1 2015-03-18 28107.  
## 2 2017-10-02 18453.  
## 3 2018-10-22 15159.

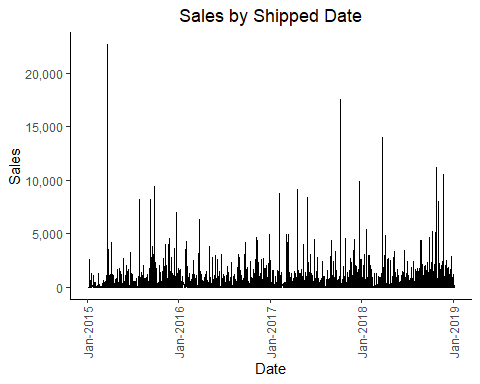
#Time with highest orders  
time\_orders <- Data %>%  
 group\_by(order\_date) %>%  
 summarize(total\_orders = length(order\_date)) %>%  
 arrange(desc(total\_orders))  
head(time\_orders,3)

## # A tibble: 3 × 2  
## order\_date total\_orders  
## <date> <int>  
## 1 2017-09-05 38  
## 2 2017-11-10 35  
## 3 2018-12-01 34

#Plot of sales by ordered date  
ggplot(Data, aes(x = order\_date, y = sales)) +  
 geom\_line() +  
 scale\_y\_continuous(labels = comma) +  
 scale\_x\_date(labels = date\_format("%b-%Y")) +  
 labs(title = "Sales by Ordered Date", x = "Date", y = "Sales") +  
 theme\_classic() +  
 theme(plot.title = element\_text(hjust = 0.5), axis.text.x = element\_text(angle = 90, vjust = 0.5))



#Plot of sales by shipped date  
ggplot(Data, aes(x = ship\_date, y = sales)) +  
 geom\_line() +  
 scale\_y\_continuous(labels = comma) +  
 scale\_x\_date(labels = date\_format("%b-%Y")) +  
 labs(title = "Sales by Shipped Date", x = "Date", y = "Sales") +  
 theme\_classic() +  
 theme(plot.title = element\_text(hjust = 0.5), axis.text.x = element\_text(angle = 90, vjust = 0.5))



Insight:

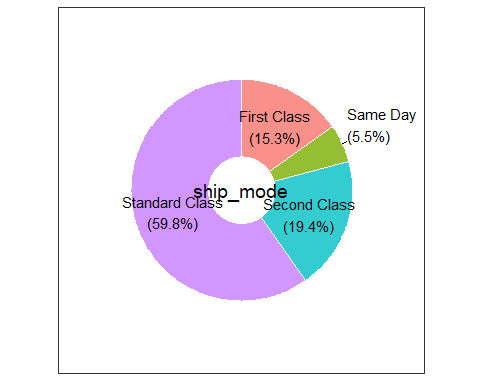
1. There was significant decrease of 4.26% in sales from 2015 to 2016.  
2. In 2018, the highest Total Sales was recorded with $722052 while 2016 had the lowest total sales of $459436.  
3. March had the highest Average Sales with $290.55 and February had the lowest Average Sales with $199.90  
4. There was a significant increase of 45.35% from February to March.  
5. There was a significant decrease of 28.61% from March to April.  
6. Over the years, November had the highest total number of orders with 1449.  
7. On 18th March 2015, recorded the highest Total Sales.  
8. On 5th Sept 2017, recorded the highest number of orders.

##### Shipping Analysis

#Shipping rankings  
shipping\_ranking <- Data %>%  
 group\_by(ship\_mode) %>%  
 summarise(orders\_shipped = length(ship\_mode)) %>%  
 arrange(desc(orders\_shipped))  
shipping\_ranking

## # A tibble: 4 × 2  
## ship\_mode orders\_shipped  
## <chr> <int>  
## 1 Standard Class 5859  
## 2 Second Class 1902  
## 3 First Class 1501  
## 4 Same Day 538

#Plot for ship mode rankings  
PieDonut(shipping\_ranking,  
 aes(ship\_mode, count = orders\_shipped),  
 title = "Shipping modes")



Insight:

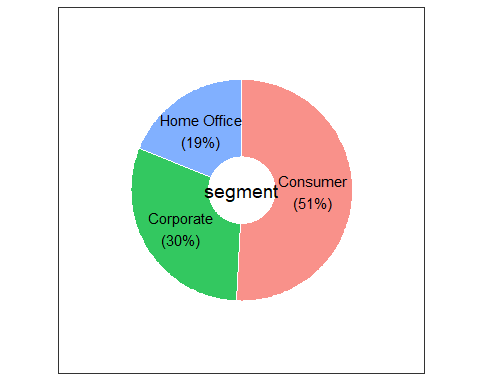
1. The most used shipping mode is 'Standard Class' by 59.8% with a count of 5859.  
2. The least used shipping mode is 'Same Day' by 5.5% with a count of 538.

##### Customer Analysis

#Segment sales  
segment\_sales <- Data %>%  
 group\_by(segment) %>%  
 summarise(total\_ordered = length(segment),  
 total\_sales = sum(sales),  
 average\_sales = mean(sales)) %>%  
 arrange(desc(total\_ordered))  
segment\_sales

## # A tibble: 3 × 4  
## segment total\_ordered total\_sales average\_sales  
## <chr> <int> <dbl> <dbl>  
## 1 Consumer 5101 1148061. 225.  
## 2 Corporate 2953 688494. 233.  
## 3 Home Office 1746 424982. 243.

#Plot of segment by sales  
PieDonut(segment\_sales,  
 aes(segment, count = total\_sales),  
 title = "Total Sales by Segment")



#Customer rankings  
customer\_ranking <- Data %>%  
 group\_by(customer\_id) %>%  
 summarise(total\_ordered = length(customer\_id)) %>%  
 arrange(desc(total\_ordered))  
#Top 3 customers  
top\_customers <- head(customer\_ranking, 3)  
top\_customers

## # A tibble: 3 × 2  
## customer\_id total\_ordered  
## <chr> <int>  
## 1 WB-21850 35  
## 2 MA-17560 34  
## 3 PP-18955 34

Insight:

1. 'Consumer' customers have the highest number of orders by 5101.  
2. 'Consumer' customers generated the highest Total Sales of 51% with $1148060.5 and the lowest Average Sales of $225.07  
3. 'Home Office' customers have the highest Average Sales with $243.40  
4. The highest number of orders by a customer is 35.

##### Geographical Analysis

#Region sales  
region\_sales <- Data %>%  
 group\_by(region) %>%  
 summarise(total\_ordered = length(region),  
 total\_sales = sum(sales),  
 average\_sales = mean(sales)) %>%  
 arrange(desc(total\_sales))  
region\_sales

## # A tibble: 4 × 4  
## region total\_ordered total\_sales average\_sales  
## <chr> <int> <dbl> <dbl>  
## 1 West 3140 710220. 226.  
## 2 East 2785 669519. 240.  
## 3 Central 2277 492647. 216.  
## 4 South 1598 389151. 244.

#Plot of region by sales  
PieDonut(region\_sales,  
 aes(region, count = total\_sales),  
 title = "Total Sales by Region")

#State sales  
state\_sales <- Data %>%  
 group\_by(state) %>%  
 summarise(total\_ordered = length(state),  
 total\_sales = sum(sales),  
 average\_sales = mean(sales)) %>%  
 arrange(desc(total\_sales))  
head(state\_sales, 10)

## # A tibble: 10 × 4  
## state total\_ordered total\_sales average\_sales  
## <chr> <int> <dbl> <dbl>  
## 1 California 1946 446306. 229.  
## 2 New York 1097 306361. 279.  
## 3 Texas 973 168573. 173.  
## 4 Washington 504 135207. 268.  
## 5 Pennsylvania 582 116277. 200.  
## 6 Florida 373 88437. 237.  
## 7 Illinois 483 79237. 164.  
## 8 Michigan 253 76136. 301.  
## 9 Ohio 454 75130. 165.  
## 10 Virginia 224 70637. 315.

#City sales  
city\_sales <- Data %>%  
 group\_by(city) %>%  
 summarise(total\_ordered = length(city),  
 total\_sales = sum(sales),  
 average\_sales = mean(sales)) %>%  
 arrange(desc(total\_sales))  
head(city\_sales, 5)

## # A tibble: 5 × 4  
## city total\_ordered total\_sales average\_sales  
## <chr> <int> <dbl> <dbl>  
## 1 New York City 891 252463. 283.  
## 2 Los Angeles 728 173420. 238.  
## 3 Seattle 426 116106. 273.  
## 4 San Francisco 500 109041. 218.  
## 5 Philadelphia 532 108842. 205.

#Mapping by state  
state\_sales\_2 <- Data %>%  
 group\_by(state) %>%  
 summarise(total\_sales = sum(sales))  
colnames(state\_sales\_2) <- c("region", "sales")  
state\_sales\_2$region <- tolower(state\_sales\_2$region)  
us\_state <- map\_data("state")  
statemap <- merge(us\_state, state\_sales\_2, by = "region")  
ggplot(statemap, aes(long, lat, group = group)) +  
 geom\_polygon(aes(fill = sales), color = "white") +  
 theme\_void() +  
 scale\_fill\_continuous(name = "Sales", low = 'lightblue', high = 'darkblue', label = comma) +  
 labs(title = "Sales by State") +  
 theme(plot.title = element\_text(hjust = 0.5))

Insight:

1. The West region generated the highest Total Sales of $710219.7 that is 31.4% while the South region had the lowest Total Sales of $389151.5 that is 17.2%.  
2. The state of California had the highest Total Sales of $446306.46 followed by New York with $306361.15  
3. New York City generated the highest Total Sales of $252462.55 and total number of orders is 1946.

### Recommendations

1. Show more ads during the months of March and April. Try special promotions or flash sales during these months to get more people interested and make conversions.  
2. Consider adding more similar products (Phones) or using similar stategies to sell more things overall.  
3. Plan your stock items and staff based on the most popular product and the months when the most people shop. Make sure you always have available products (especially products that sold the most) and your team is ready to help customers during these times.  
4. Leverage high March sales to plan exciting marketing campaigns. Think about special promotions or events that customers will love during this time.