Homework - 3

Team - 18

11/2/2019

Contents

Problem and Approach **2**

Our Task . . . 2

What is Success? . . . 2

Data Cleaning, Trasformation and Feature Engineering **2**

Data Understanding and Availability . . . 2

Analysis **4**

Understanding the Current Situations . . . 4

Analyzing Hourly Sales . . . 5

Weekday - Weekend Split . . . 6

Item Analysis . . . 7

Analyzing Coffee Sales . . . 10

Food Category . . . 12

Extras Category . . . 12

Non-Caffeinated . . . 14

Consumer Buying Patterns . . . 15

Analysis . . . 15

Rules for Frequently Bought Items . . . 16

Rules for Frequently Bought Categories . . . 16

Findings: . . . 16

Customer Loyalty Analysis . . . 17

Approach . . . 17

Analysis . . . 18

Findings . . . 20

Final Takeaways **20**

Trend Analysis . . . 20

Cluster Analysis . . . 21

Purchase Behavior Analysis . . . 21

Recommendations **22**

Problem and Approach

Our Task

Central Perk is a boutique coffee shop in New York City that has decided to take a data-driven approach to assess the current state of their operations and customer base. To assist them in doing that, we've been hired as a consultant team to perform exploratory analysis on the data provided.

At the moment, Central Perk believes that they have a sense of their customer base, their current customers are fairly loyal and business is consistent year over year. Their belief in customer loyalty is such that they're not necessarily interested in acquiring new customers but would rather focus on increasing revenue from loyal customers.

To provide better business insights to Central Perk, our analysis will focus on three angles:
1. Identify the buying trends for each category of our customers.
2. Generate further insights about our customers, such as loyalty, using the data provided.
3. Analyze the sale of products at Central Perk and their relationship to each other.

What is Success?

After our analysis of Central Perk's problem of interest, our goal is to identify distinct groups of customers and develop targeting strategies to further increase the overall revenue for Central Perk. We analyze store operations, sales, and consumers to formulate these strategies.

Data Cleaning, Trasformation and Feature Engineering

Data Understanding and Availability

The team has been proveded with three data files for the years 2016, 2017, and 2018. The data is at an transactional level, we have date and time for the transactions, items, and category. To determine Central Perk's revenue, we use gross sales which is Net sales after discounts. Customer ID contains null field which are assumed to be transactions by customers without store credit card/memberships.

Importing Libraries

```r
library(tidyverse)
library(lubridate)
library(reshape2)
library(splitstackshape)
library(arules)
library(reshape)
library(scales)
```

Dataset Import

We check the date ranges for each of the files

```r
sales_2016$Date <- as.Date(sales_2016$Date, format = '%m/%d/%y')
sales_2017$Date <- as.Date(sales_2017$Date, format = '%m/%d/%y')
sales_2018$Date <- as.Date(sales_2018$Date, format = '%m/%d/%y')
```

```
cat(paste("Min_2016: ", min(sales_2016$Date), " Max_2016: ", max(sales_2016$Date),
          "\nMin_2017: ", min(sales_2017$Date), " Max_2017: ", max(sales_2017$Date),
          "\nMin_2018: ", min(sales_2018$Date), " Max_2018: ", max(sales_2018$Date)))
```

```
## Min_2016:  2017-01-01  Max_2016:  2017-12-31
## Min_2017:  2016-07-15  Max_2017:  2016-12-31
## Min_2018:  2018-01-02  Max_2018:  2018-08-24
```

It seems like 2016 file contains 2017 data. Going ahead, we swap the names of the files for 2016 and 2017.

```
data_16 <- sales_2017
data_17 <- sales_2016
data_18 <- sales_2018
```

We can also observe that We have data from **15th July 2016** to **24th August 2018**.
We further observe that the net sales column contains negative values and also dollar signs. We clean these and convert this column to integers.

```
# Weekly Sales totals
data_16$Net_Sales <- as.numeric(gsub("\\$", "", data_16$Net.Sales))
data_17$Net_Sales <- gsub("\\$", "", data_17$Net.Sales)
data_17$Net_Sales <- gsub("\\(", "", data_17$Net_Sales)
data_17$Net_Sales <- gsub("\\)", "", data_17$Net_Sales)
data_17$Net_Sales <- as.numeric(data_17$Net_Sales) * data_17$Qty / abs(data_17$Qty)
data_18$Net_Sales <- as.numeric(gsub("\\$", "", data_18$Net.Sales))
```

We create time based sub-columns such as week, day, month, hour and combine the dataframes which are required for the analysis. We Ignore the null Customer IDs for now, as we are only interested in all customer transactions.

```
data_16$week <- week(data_16$Date)
data_17$week <- week(data_17$Date)
data_18$week <- week(data_18$Date)

data_16$month <- month(data_16$Date)
data_17$month <- month(data_17$Date)
data_18$month <- month(data_18$Date)

data_16$year <- year(data_16$Date)
data_17$year <- year(data_17$Date)
data_18$year <- year(data_18$Date)

data_16$weekday <- weekdays(data_16$Date)
data_17$weekday <- weekdays(data_17$Date)
data_18$weekday <- weekdays(data_18$Date)

data_16$hour <- hour(strptime(data_16$Time, format = "%H:%M:%S"))
data_17$hour <- hour(strptime(data_17$Time, format = "%H:%M:%S"))
data_18$hour <- hour(strptime(data_18$Time, format = "%H:%M:%S"))

data = rbind(data_16, data_17, data_18)
data$weekend_flag <- ifelse(data$weekday == "Saturday" | data$weekday == "Saturday",
                            "Weekend", "Weekday")

data$DateTime <- paste(data$Date, data$Time)
data$Item<- ifelse(data$Item == 'ðŸ\u008d‹LemonadeðŸ\u008d‹', "Lemonade", data$Item)
```

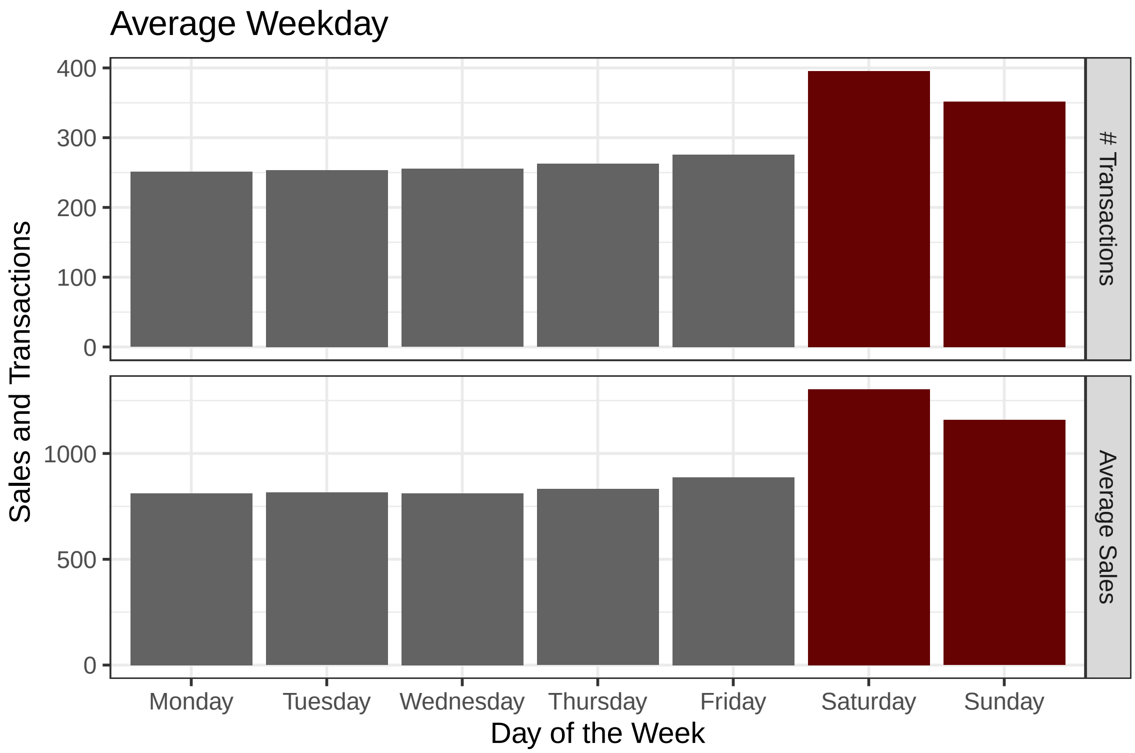
Analysis

Understanding the Current Situations

We first analyze the current sales trends for the coffee shop. As we have observed in the above data cleaning section, we have transactional data from July 2016 to August 2018. We first analyze total monthly transactions, number of distinct customers, and sales across all of the data. We are interested in monthly, weekly, and daily trends and also top selling categories/items. This can be leveraged to increase prices and offer discounts at certain times which can normalize demand.

We focus this analysis to improve day-to-day operations in the cafe. We look at weekly, daily, and hourly operations to generate granular insights. Looking at the number of transactions atand average sales at a weekday level:

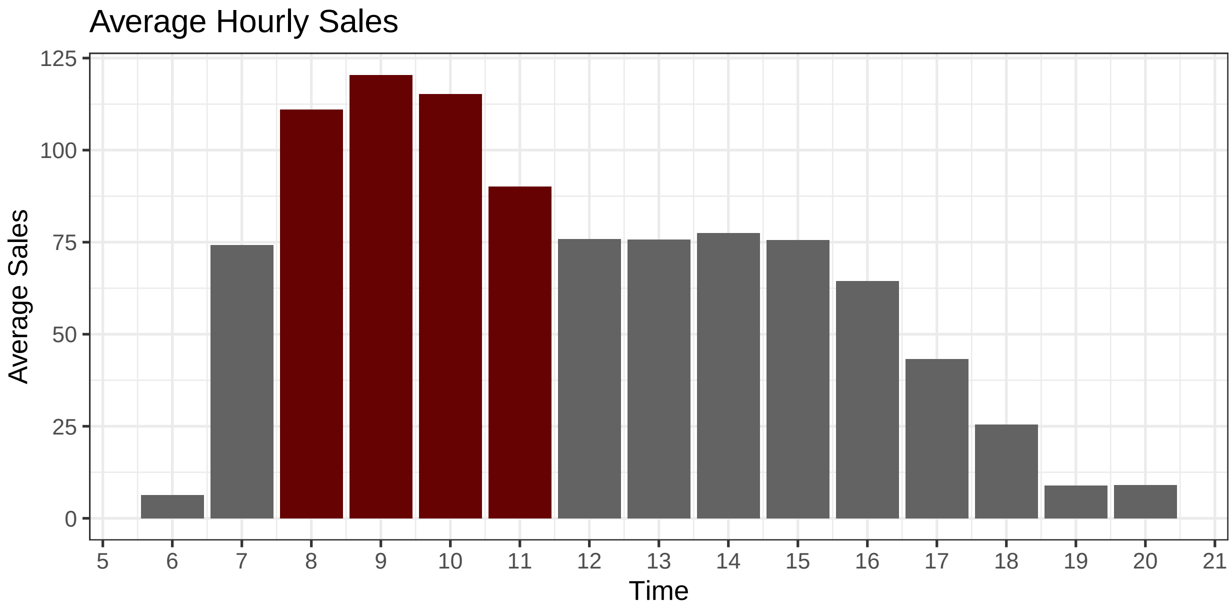
```
weekday_data <- data %>%
  group_by(year, month, week, weekday) %>%
  summarise(weekday_sales = sum(Net_Sales, na.rm = T),
            weekday_transactions = n()) %>%
  ungroup() %>%
  group_by(weekday) %>%
  summarise(avg_sales = mean(weekday_sales),
            num_transactions = mean(weekday_transactions)) %>%
  mutate(a = "Average Sales",
         b = "# Transactions",
         flg = ifelse((weekday == "Saturday" | weekday == "Sunday" ), 1, 0))

avg_sls <- weekday_data %>% select(weekday, avg_sales, a, flg) %>%
  dplyr::rename(label = a, metric = avg_sales)
num_tx <- weekday_data %>% select(weekday, num_transactions, b, flg) %>%
  dplyr::rename(label = b, metric = num_transactions)
weekday_data <- rbind(avg_sls, num_tx)

days <- c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday","Sunday")
ggplot(weekday_data, aes(x = weekday, y = metric, fill = flg)) +
  scale_x_discrete(limits = days) +
  geom_bar(stat = 'identity',
           fill = c(rep("#636363", 5),rep("#660202", 2),
                    rep("#636363", 5),rep("#660202", 2))) +
  labs(title = "Average Weekday", x = "Day of the Week", y = "Sales and Transactions ") +
  theme_bw() +
  theme(legend.position="none") +
  facet_grid(label~., scales = "free")
```



We observe that there is a clear distinction between weekdays ans weekends. Weekdays experience lower amount of traffic as well as Sales as compared to weekends. Weekdays and weekends show a difference in behavior. Weekends have an average of 2x the transactions to that of weekdays. So we divide our further analysis into weekdays and weekends.

Analyzing Hourly Sales

```r
hour_data <- data %>%
  group_by(year, month, week, weekday, hour) %>%
  summarise(hour_sales = sum(Net_Sales, na.rm = T)) %>%
  ungroup() %>%
  group_by(hour) %>%
  summarise(avg_sales = mean(hour_sales))

ggplot(hour_data, aes(x = hour, y = avg_sales)) +
  geom_bar(stat = 'identity',
           fill = c(rep("#636363", 2),rep("#660202", 4), rep("#636363", 9))) +
  labs(title = "Average Hourly Sales", x = "Time", y = "Average Sales") +
  scale_x_continuous(breaks = c(0:24)) +
  theme_bw()
```

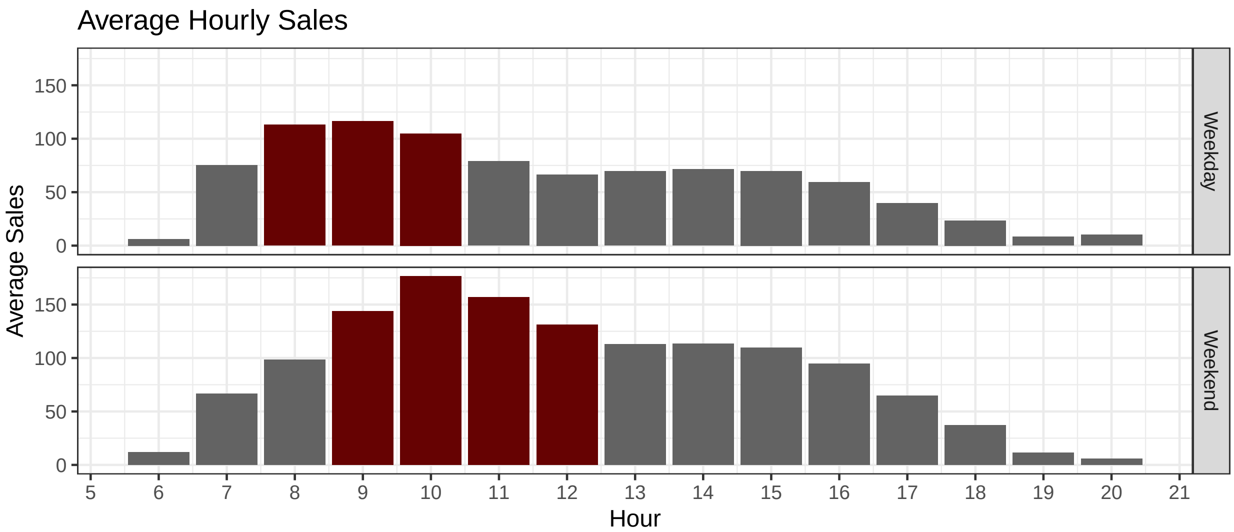


It is clearly observed that Central Perk experiences a surge in sales between **8 AM** to **12 AM** in the morning. The store experiences more than **$75** of sales in this time which can be considered as normal cutoff. We can classify this into rush hour and not rush hour traffic. The coffee shop seem to be experiencing high traffic during this time, which needs to be normalized as per them.

Weekday - Weekend Split

We further split this into weekdays and weekends to examine the pattern betweeen the two, which can also tell is the rush hour problem needs to be tackeled differently for both.

```r
flag_hour_data <- data %>%
  group_by(year, month, weekend_flag, week, weekday, hour) %>%
  summarise(hour_sales = sum(Net_Sales, na.rm = T)) %>%
  ungroup() %>%
  group_by(weekend_flag, hour) %>%
  summarise(avg_sales = mean(hour_sales))

ggplot(flag_hour_data, aes(x = hour, y = avg_sales)) +
  geom_bar(stat = 'identity', position = 'dodge',
           fill = c(rep("#636363", 2),rep("#660202", 3), rep("#636363", 10),
                    rep("#636363", 3),rep("#660202", 4), rep("#636363", 8))) +
  labs(title = "Average Hourly Sales", x = "Hour", y = "Average Sales") +
  scale_x_continuous(breaks = c(0:24)) +
  facet_grid(weekend_flag~.) +
  theme_bw()
```



There we go. There is a clear distinction between the two. Not only the rush hour vs not rush hour cutoff is different, also the times when this happens also changes accordingly. For weekends the rush hour changes from **9 AM to 12PM** in the morning and the average net sales for non rush hour also are **$100**.

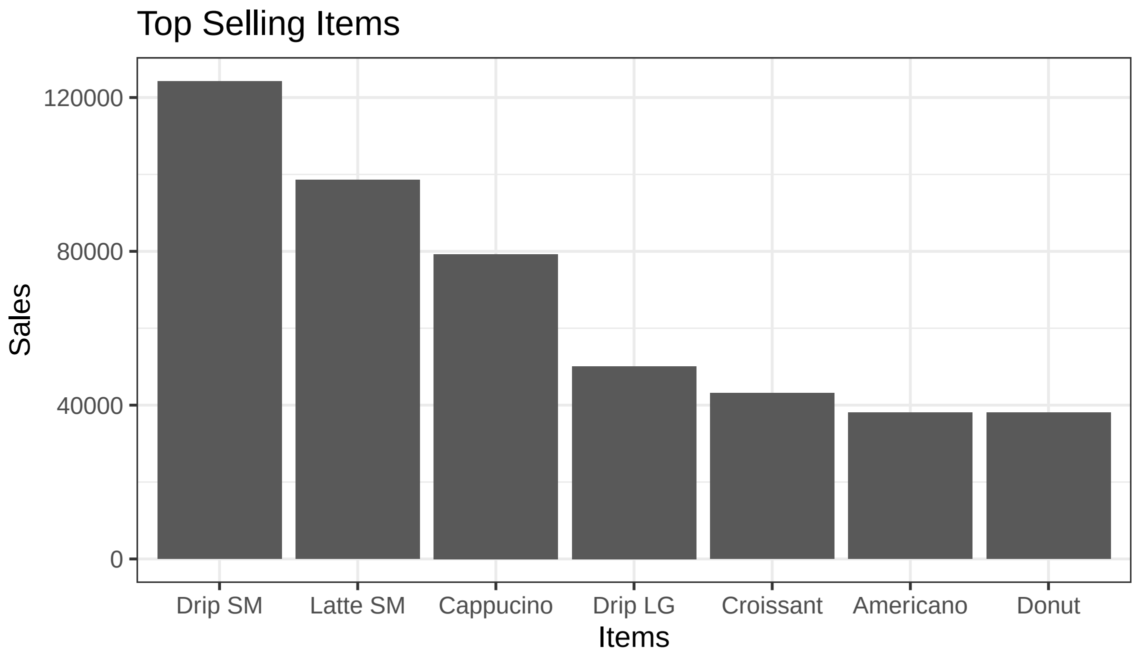
Weekdays observe maximum traffic from **8 AM - 11 AM** and weekends observe max traffic from 9 AM to 12 AM. Prices can be hiked during these times taking advantage of the rise in traffic. Both weekdays and weekends experience a dip in traffic after 4 PM in the evening where discounts can be offered to consumers which will clear inventory faster, get more people in the door, and also lead to increase in revenue

Now that we have clear distinction between sales and transaction patterns on weekdays and weekends, we further deep dive into items which are sold frequently together.

Item Analysis

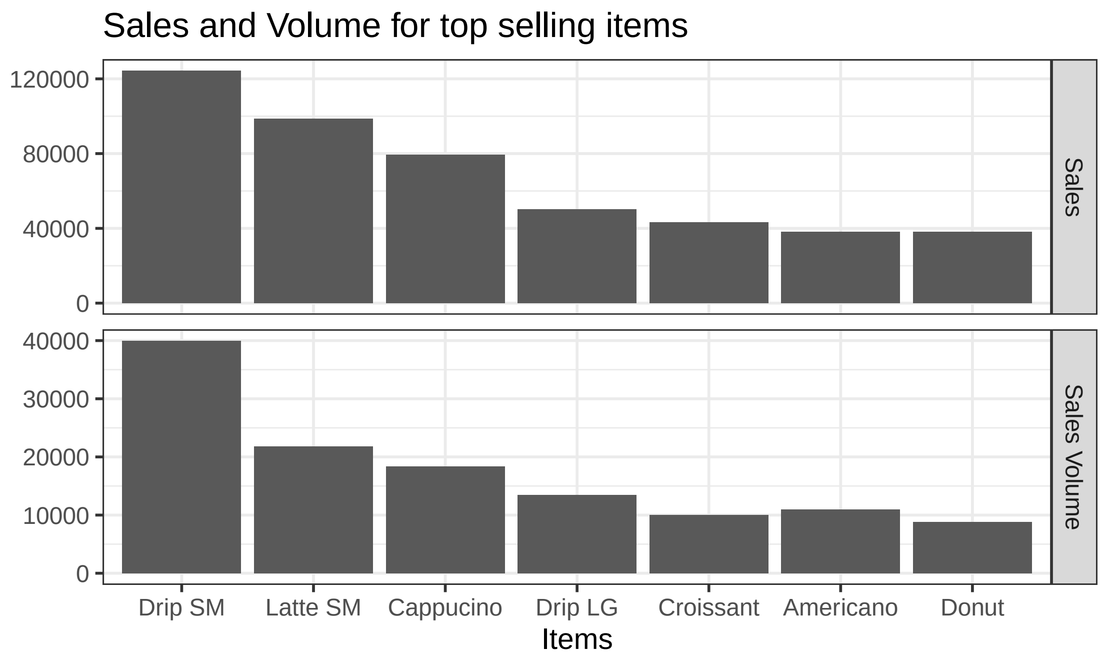
We observe the top selling products by net sales

```
data <- data %>% filter(Item != '')

salesI <-data %>%
  group_by(Item) %>%
  summarise(sales = sum(Net_Sales, na.rm = TRUE)) %>%
  arrange(desc(sales)) %>%
  slice(1: 7)

salesI$Item <- factor(salesI$Item, levels = salesI$Item[order(-salesI$sales)])

ggplot(salesI, aes(x=Item, y=sales)) +
  geom_bar(stat='identity', position='dodge') +
  labs(title = "Top Selling Items", y = "Sales", x = "Items") +
  theme_bw()
```



We find that there are five kinds of coffee (Drip SM, Latte SM, Cappucino, Drip LG and Americano) and two kinds of snacks (Croissant, Donut) which are sold the most. If we also look at the number of items sold, we see a similar trend.

```r
vs <-data %>%
  group_by(Item) %>%
  summarise(num = n(), sales = sum(Net_Sales, na.rm = TRUE)) %>%
  arrange(desc(sales)) %>%
  slice(1: 7) %>%
  mutate(a = "Sales Volume",
         b = "Sales")
a1 <- vs %>% select(Item, num, a) %>%
  dplyr::rename(metric = num, label = a)
b1 <- vs %>% select(Item, sales, b) %>%
  dplyr::rename(metric = sales, label = b)

vs1 <- rbind(a1,b1)

itm <- c("Drip SM", "Latte SM", "Cappucino",
         "Drip LG", "Croissant", "Americano", "Donut")
ggplot(vs1, aes(x=Item, y=metric)) +
  geom_bar(stat='identity', position='dodge') +
  scale_x_discrete(limits = itm) +
  facet_grid(label~., scale = "free") +
  labs(title = "Sales and Volume for top selling items", x = "Items", y = "") +
  theme_bw()
```

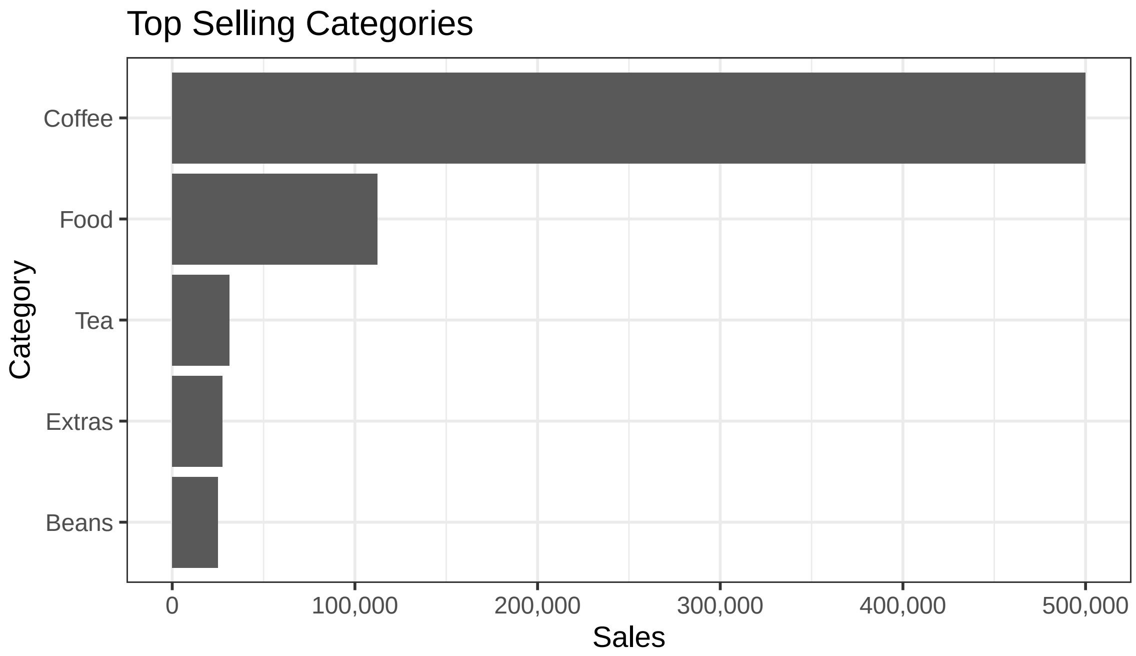


To better understand the sales distribution, we explore the net sales based on category. On examining the top categories, we again see that Coffee, Extras, and Food are the top three categories, and the sales are mainly driven be coffee. We proceed to further analyze these three categories similar to above analysis. We look at hourly sales as well as weekdays vs weekends to detect similarities in coffee consumptions and also find anomalies, if any.

```
salesC <-data %>%
  filter(Category != 'None') %>%
  group_by(Category) %>%
  summarise(sales = sum(Net_Sales, na.rm = TRUE)) %>%
  top_n(n = 5)
```

```
## Selecting by sales
```

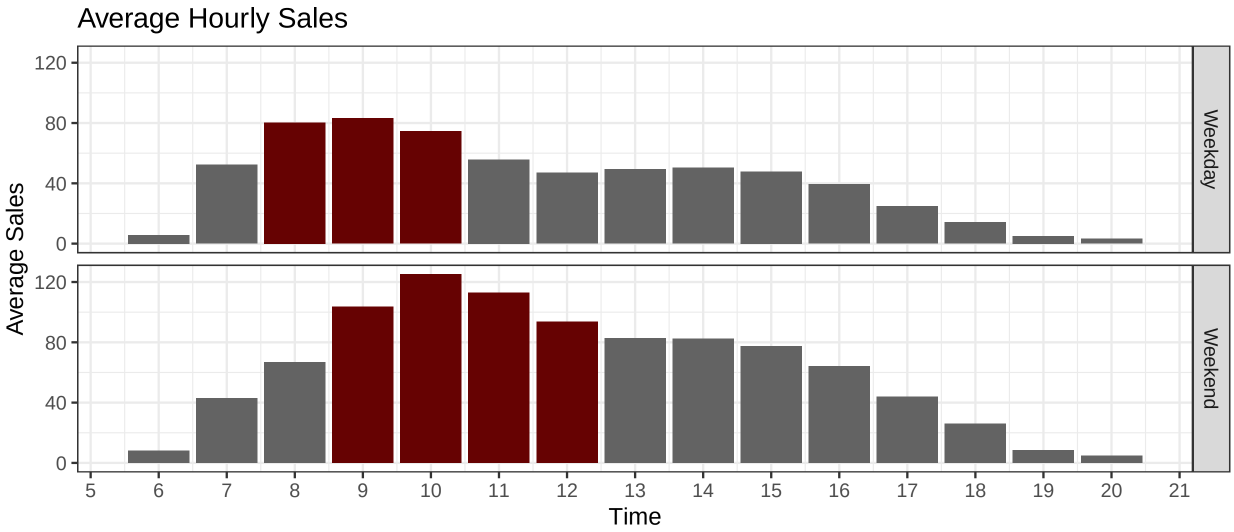
```
salesC$Category <- factor(salesC$Category,
                          levels = salesC$Category[order(salesC$sales)])

ggplot(salesC, aes(x=Category, y=sales)) +
  geom_bar(stat='identity', position='dodge') +
  labs(title = "Top Selling Categories", y = "Sales") +
  theme_bw() +
  scale_y_continuous(labels = comma) +
  coord_flip()
```



Analyzing Coffee Sales

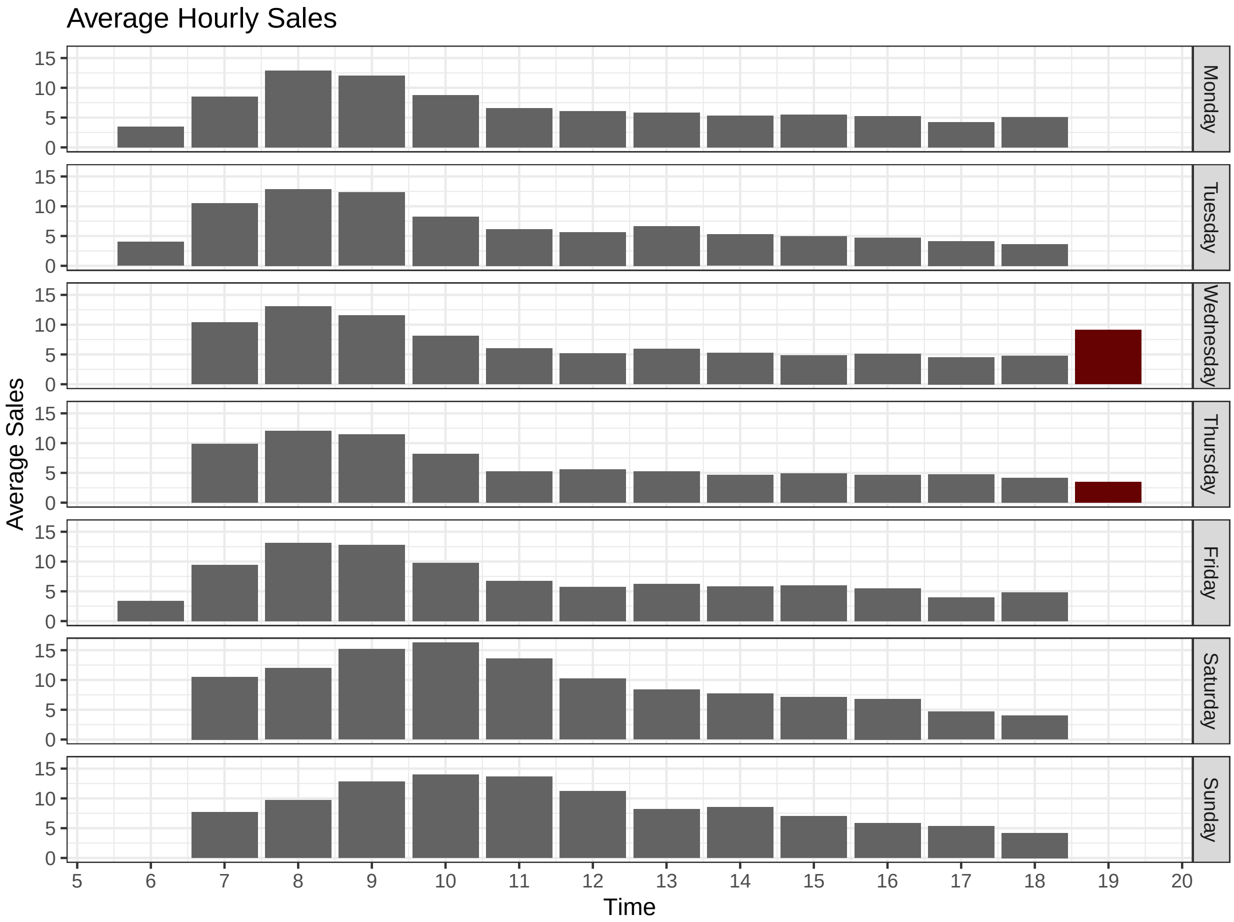
After that, we further dive into top categories and top selling items in these categories. Since weekday vs weekend is our key distinguishing factor, we observe coffee sales based on these two facets. We find that coffee sales are higher on weekends, especially on Saturdays. The peak hour is around 8 AM and 9 AM in morning, and the sales drop slightly between 12 PM and 1 PM at which time people are usually having lunch and continue to drop after a slight recovery. Besides, the peak hour on weekend is around 10 AM, few hours later than weekdays. The dip observed during weekdays during lunch time, is not observed during weekends.

```r
hour_data <- data %>%
  filter(Category == 'Coffee') %>%
  group_by(year, month, week,weekend_flag, weekday, hour) %>%
  summarise(hour_sales = sum(Net_Sales, na.rm = T)) %>%
  ungroup() %>%
  group_by(weekend_flag, hour) %>%
  summarise(avg_sales = mean(hour_sales))

ggplot(hour_data, aes(x = hour, y = avg_sales)) +
  geom_bar(stat = 'identity',
           fill = c(rep("#636363", 2),rep("#660202", 3), rep("#636363", 10),
                    rep("#636363", 3),rep("#660202", 4), rep("#636363", 8))) +
  labs(title = "Average Hourly Sales", x = "Time", y = "Average Sales") +
  scale_x_continuous(breaks = c(0:24)) +
  facet_grid(weekend_flag~.)+
  theme_bw()
```



Analyzing each of the above categories and their items for hourly sales for weekdays vs weekend, we do observe an anomaly in **Drip LG**. There is a spike in sales observed on Wednesdays and Thursdays after **6 PM**.

```
weekday_hour_data <- data %>%
  filter(Item == 'Drip LG') %>%
  group_by(year, month, week, weekday, hour) %>%
  summarise(hour_sales = sum(Net_Sales, na.rm = T)) %>%
  ungroup() %>%
  group_by(weekday, hour) %>%
  summarise(avg_sales = mean(hour_sales)) %>%
  mutate(flag = ifelse(((weekday == "Wednesday" | weekday == "Thursday") & (hour == 19)),
                       1, 0))

weekday_hour_data$weekday_f <- factor(weekday_hour_data$weekday,
                               levels = c("Monday", "Tuesday", "Wednesday",
                                          "Thursday", "Friday", "Saturday","Sunday"))
ggplot(weekday_hour_data, aes(x = hour, y = avg_sales, fill = factor(flag))) +
  geom_bar(stat = 'identity',
           fill = c(rep("#636363", 38),rep("#660202", 1),
                    rep("#636363", 12), rep("#660202", 1),
                    rep("#636363", 37))) +
  labs(title = "Average Hourly Sales", x = "Time", y = "Average Sales") +
  scale_colour_manual(values=c('black', 'red')) +
  scale_x_continuous(breaks = c(0:24)) +
  facet_grid(weekday_f~.)+
  theme_bw() +
  theme(legend.position="none")
```



Upon further inspection, we observe that the store has a spike in sales of **$9.50** USD on Wednesday and **$3.50** USD on Thursday. This can be extended to other weekdays as well if the cost of keeping the store open for an additional hour is less than the average sales.

Food Category

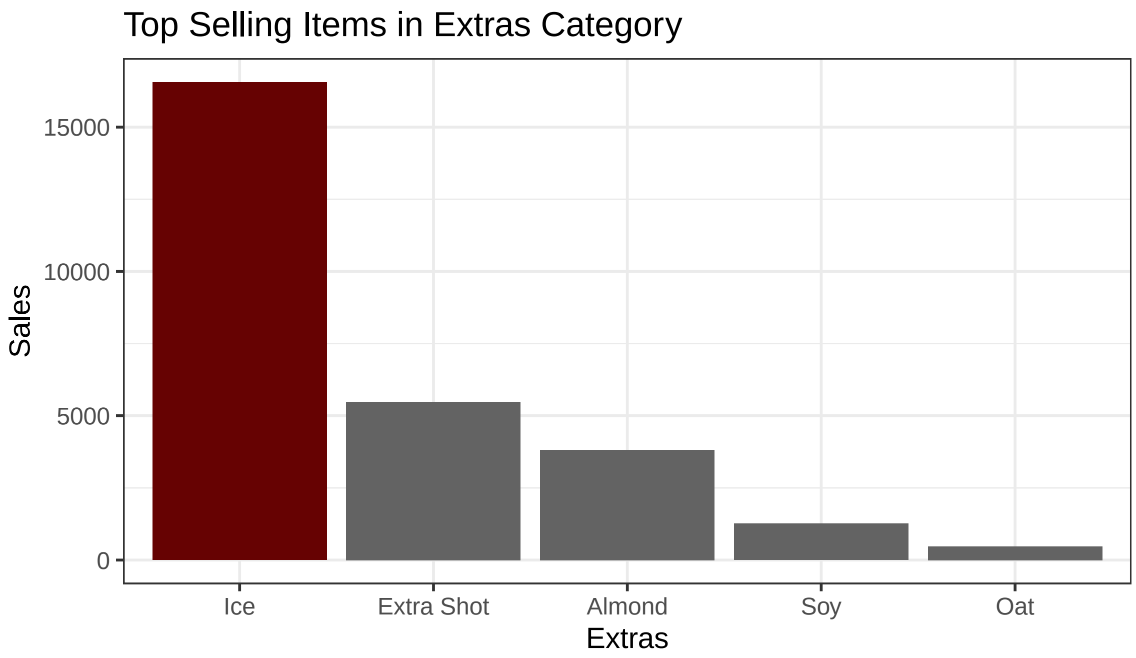
On performing a similar analysis for the food category, we observa a similar trend as coffee with an exception of the peak hours. There is a distinction between weekdays and weekends, the peak hours start at **7 AM** on Weekdays instead of **8 AM**. The hourly trends have no anomalies or outliers.

Extras Category

To better understand the sales of other items in Central Perk, we also explore the sales and sale volume of other categories, and we obtain two interesting findings.

```
extras_s <-data %>%
  filter(Category == 'Extras') %>%
  group_by(Item) %>%
  summarise(sales = sum(Net_Sales, na.rm = TRUE))

extras_s$Item <- factor(extras_s$Item, levels = extras_s$Item[order(-extras_s$sales)])
```

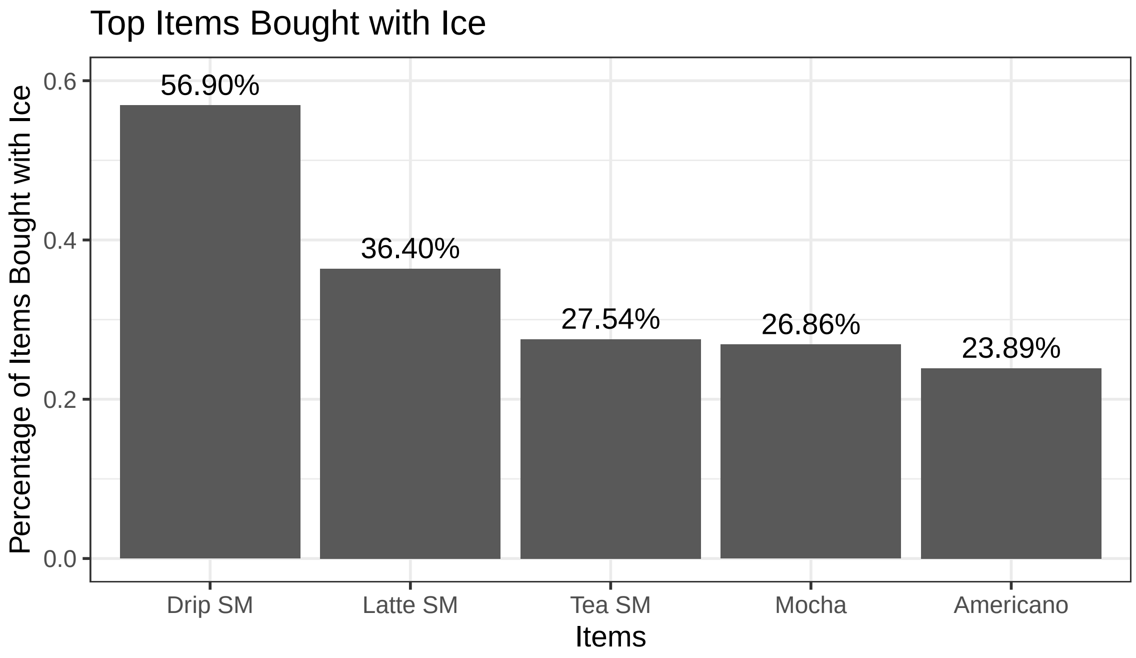
```r
ggplot(extras_s, aes(x=Item, y=sales)) +
  geom_bar(stat='identity', position='dodge',
           fill = c(rep("#660202", 1),rep("#636363", 4))) +
  labs(title = "Top Selling Items in Extras Category", x = "Extras", y = "Sales" ) +
  theme_bw()
```



We find that ice drives the sales of Extras category. We curisously deep dive into what ice is bought with as an extra and got an interesting finding.

We only filtered the transactions which contained ice. **56.9%** times people bought Small Drip Coffee, they bought ice as an extra. Besides, people also would like to bought ice when they bought Latte LM, Tea SM, Mocha, and Americano. These items could be pre-bundled with ice, which can drive up the overall sales for the coffee shop.

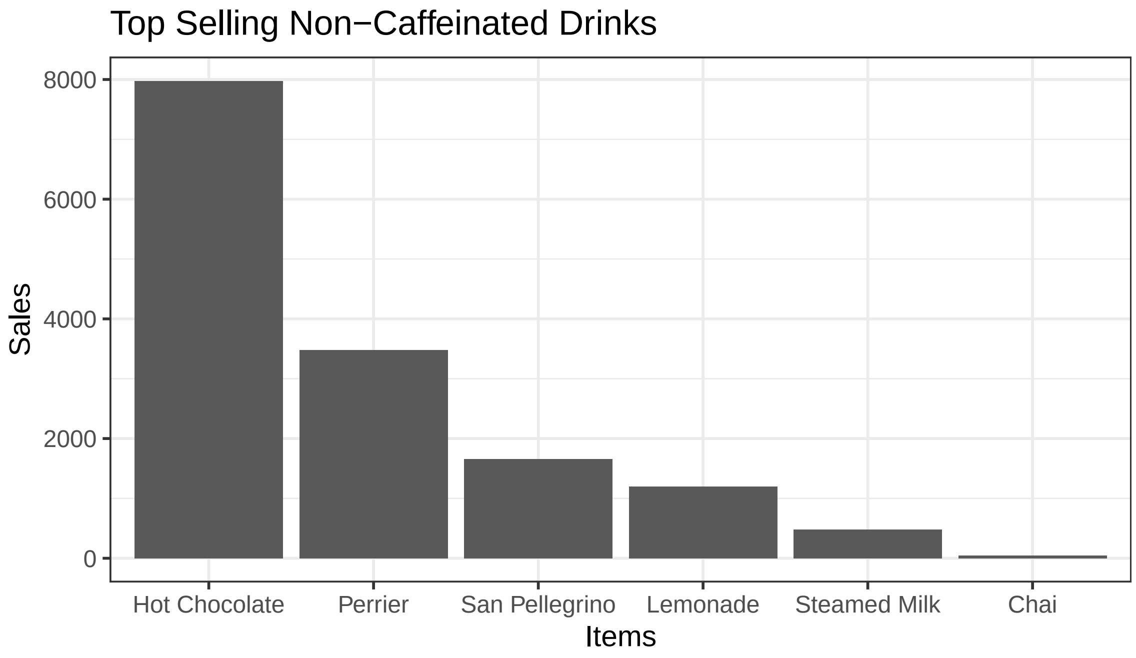
```r
dd <- data %>% select(Date, Time, Category, Item, Customer.ID)
ice <- dd %>%
  filter(Item == 'Ice') %>%
  left_join(dd, by = c('Date', 'Time'))

ice2 <- ice %>% filter(!(Item.x == 'Ice' & Item.y == 'Ice')) %>%
  select(Item.x, Item.y) %>%
  group_by(Item.x, Item.y) %>%
  summarise(count = n())

item_vol <- data %>% select(Item) %>%
  group_by(Item) %>%
  summarise(count = n())

ice3 <- ice2 %>% left_join(item_vol, by = c('Item.y' = 'Item')) %>%
  mutate(percent = count.x / count.y) %>%
  arrange(desc(percent)) %>%
  slice(1: 7) %>%
  filter(Item.y != 'Oat' & Item.y != 'Extra Shot')

ice3$Item.y <- factor(ice3$Item.y,
                      levels = ice3$Item.y[order(-ice3$percent)])

ggplot(ice3, aes(x=Item.y, y=percent)) +
  geom_bar(stat='identity', position='dodge') +
  labs(title = "Top Items Bought with Ice",x = 'Items',
       y = 'Percentage of Items Bought with Ice') +
  geom_text(aes(label = sprintf("%1.2f%%", 100*percent)),
            position = position_dodge(width = 1),
            vjust = -0.5, size = 4) +
   theme_bw() +
   ylim(0.0, 0.6)
```



Non-Caffeinated

Apart from coffee, there are also many kinds of non-caffein drinks at Central Perk. Chocolate and Perrier are the top two items sold in these category.

```
Non_CaffeinatedDrinks_s <-data %>%
  filter(Category == 'Non-Caffeinated Drinks') %>%
  group_by(Item) %>%
  summarise(sales = sum(Net_Sales, na.rm = TRUE)) %>%
  ungroup()

Non_CaffeinatedDrinks_s$Item <- factor(Non_CaffeinatedDrinks_s$Item,
levels = Non_CaffeinatedDrinks_s$Item[order(-Non_CaffeinatedDrinks_s$sales)])

ggplot(Non_CaffeinatedDrinks_s, aes(x=Item, y=sales)) +
  geom_bar(stat='identity', position='dodge') +
  labs(title = "Top Selling Non-Caffeinated Drinks", x = "Items", y = "Sales") +
  theme_bw()
```



Consumer Buying Patterns

Now we proceed to apply association rules to find which items are frequently bought together. Consumer tend to have biases towards specific products which can differ by area and demographics which can be hard to detect by intuition. We use association rules in such cases which can identify what items occur together in a transaction. For every possible combination of a transaction data, associaition rules can find how many times a particular combination has occured, how much does purchase of one entity lead to purchase of the other, and also by how much this purchase combination compared just random chance.

We cannot only use this to find items which sell together, but also to find categories which consumers perfer to buy together as well.

Analysis

We transform the data to represent a transaction on a single row and then converting these to a transaction file to apply the apriori algorithm.

```
cols <- c( "DateTime", "Item")

Transactions_Data <- data[cols]

# Take Unique Items bought together
Transactions_Data <- Transactions_Data %>%
  group_by(DateTime, Item)%>%
  summarise(c = n()) %>%
  select(DateTime, Item)

# What are the max items per transaction?
max_items <- Transactions_Data %>%
  group_by(DateTime) %>%
  summarise(count = n()) %>%
  arrange(desc(count))

grp_Transactions <- Transactions_Data %>%
  group_by(DateTime)%>%
  mutate(Items = paste0(Item, collapse = ",")) %>%
  ungroup() %>%
  group_by(DateTime, Items) %>%
  summarise(c = n()) %>%
  ungroup()%>%
  select(Items)

write.csv(grp_Transactions, "transactions.csv", quote = FALSE, row.names = TRUE)
txn = read.transactions(file="transactions.csv",
                        rm.duplicates= TRUE, format="basket",sep=",",cols=1)

txn@itemInfo$labels <- gsub("\"","", txn@itemInfo$labels)
basket_rules <- apriori(txn,parameter = list(sup = 0.01, conf = 0.05,  target="rules"))
inspect(sort(basket_rules, by = "lift"))
```

Rules for Frequently Bought Items

Assocuiation Rules	Support	Confidence	Lift	Count
Lattee -> Almond	0.03	0.53	3.3	3889
Drip Small –>Ice	0.15	0.64	2.2	19571
Cappucino –> Almond	0.012	0.09	1.56	1576
Latte Small -> Ice	0.05	0.3	1.38	6703

Rules for Frequently Bought Categories

Assocuiation Rules	Support	Confidence	Lift	Count
Coffee –> Extras	0.3	0.94	2.06	38486
Coffee –> {Food, Extras}	0.04	0.93	2.1	5765
Cappucino –> Almond	0.012	0.09	1.9	1576

Findings:

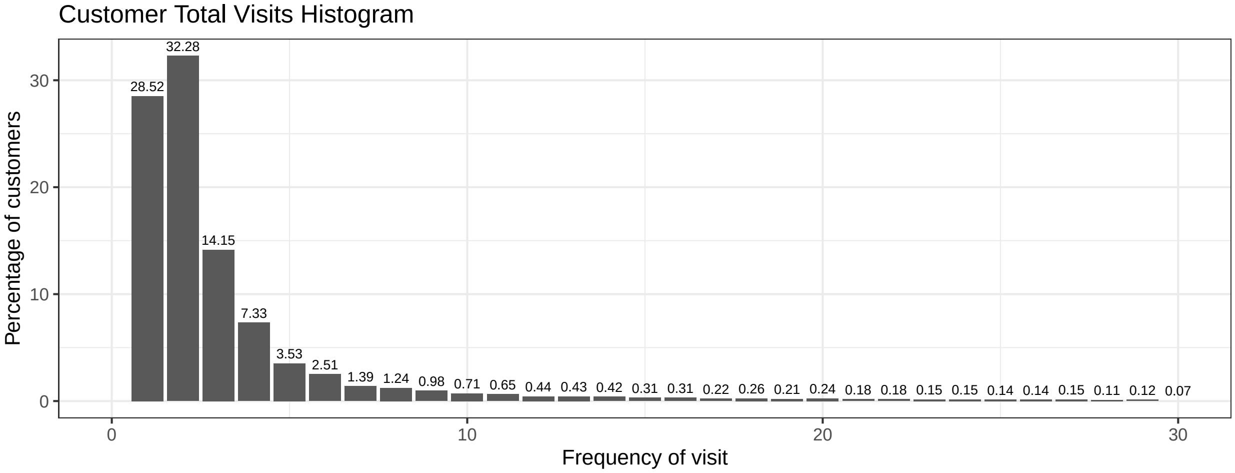
1. Coffee and Food are the most frequently occuring categories. These can either be bundle together or either of them can be marked up to normalize the demand.
2. Almond seems to be the go-to extra which is mostly consumed with Latte and Drip coffee.

Customer Loyalty Analysis

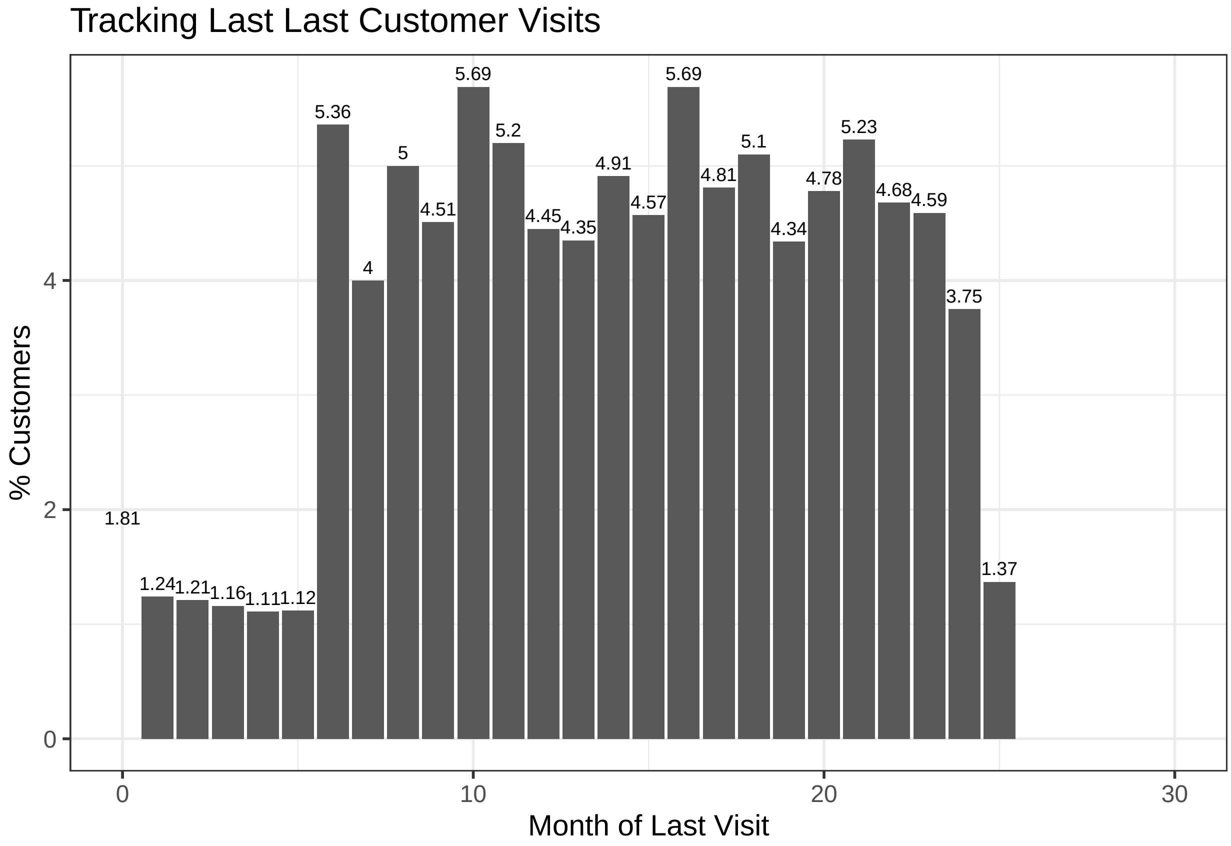
Approach

Central Perk has a belief that their customer base is loyal. Thus, their interests lie in increasing the revenue generated from these "loyal" customers over acquiring new customers. To address this belief, we analyzed customers having a Central Perk membership (a customer ID) for at least 6 months and measured their value in recency and frequency.

Recency is defined as the number of months between a customer's most recent purchase and the latest date in the dataset, which is 2018-08-24. Frequency is defined as the number of visits a particular customer made to the store through the two year period. We used these parameters to understand customer behavior.



```r
ggplot(data=recency_new, aes(x=recency_months, y=perc)) +
  geom_bar(stat="identity")+ xlim(0, 30) +
  geom_text(aes(label=perc), vjust=-0.5, size= 2.5) +
  labs(title = 'Tracking Last Last Customer Visits',
       x = "Month of Last Visit", y = '% Customers') +
  theme_bw()
```



From the two graphs, we can see that only 25% of customers visited more than 3 times in their lifetime. Only 6% of customers repeated their transaction in the last 6 months. This clearly contradicts Central Perk's belief on their high customer loyalty. Thus the team applied the following cluster analysis to generate data-driven consumer segmentation.

Analysis

To prepare data for our analysis, the transaction level data was transformed into customer level data based on Customer ID. We removed any transaction data that did not contain a Customer ID due to the lack of information to uniquely identify those consumers.

We then segmented the remaining customers using four values:
1. Total Spending
2. Total Number of Visits
3. Number of Days Since Their Last Visit
4. Visit Frequency (number of visits since their first visit)

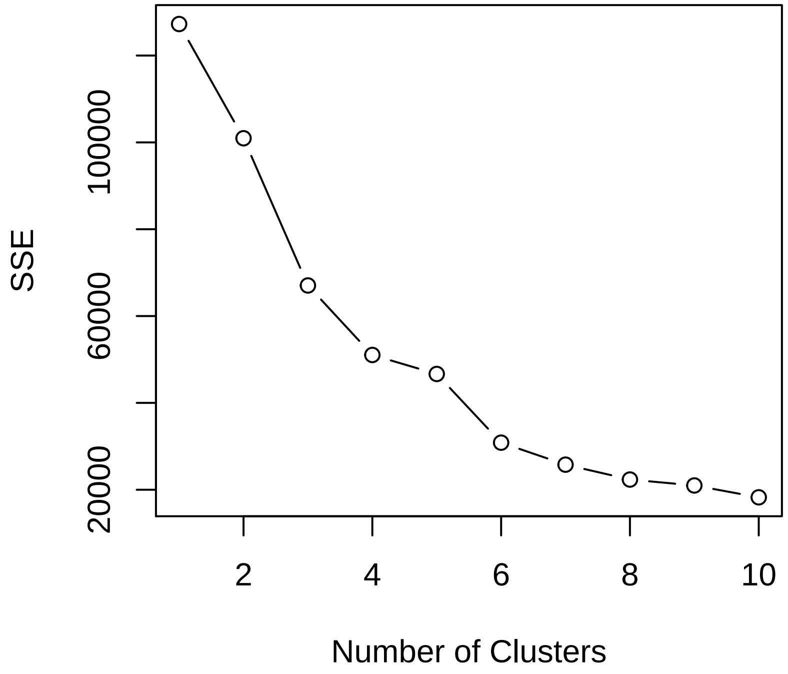
To prepare the data for clustering via k-means, we normalized the variables using min-max scaling so that outliers do not affect the variables which are more centrally distributed in the data.

This was done with the following code:

We first plot the SSE curve to find optimal number of clusters:

```
set.seed(123)
SSE_curve <- c()
```

```
for (k in 1:10) {
  kcluster <- kmeans(df_2[-c(1)], k)
  sse <- sum(kcluster$withinss)
  SSE_curve[k] <- sse
}
plot(1:10, SSE_curve, type="b", xlab="Number of Clusters", ylab="SSE")
```



Looking at the SSE curve above, we find that the optimal number of clusters is **four**. We proceed to make four clusters from the data, and then analyze them to find characteristics. The code to create the four clusters is shown below:

```
kcluster <- kmeans(df_2[-c(1)], 4)
```

Finally, a characteristic summary of the four cluster is provided below:

cluster	spent	recency	visit	Visit frequency	days_since_first_visit	count
1	11.9	163.9	3.7	0.0335	213.1	16373
2	9.0	563.1	2.7	0.004	576.0	14811
3	156.6	114.0	47.6	0.486	370.4	574
4	815.0	87.7	239.9	0.425	646.8	58

Findings

Based on the summary, we combine clusters 3 and 4 together because compared to other clusters, these two clusters show similar trends in general with high average spend, lower recency, and high visit frequency. Additionally, we added one extra variable, days since their first visit, to evaluate each cluster.

From this, we were able to identify the three different types of customers:
1. Loyal 2. Newcomers 3. Not Loyal

Loyal:

Compared to Newcomers and Not Loyal, Loyal has the highest average spending, the lowest recency, and the highest total visit and visit frequency. Thus we identified them as our loyal customers.

cluster	spent	recency	visit	Visit frequency	days_since_first_visit	count
Loyal	217	112	65	0.4805	396	632

Newcomers:

For Newcomers, days since their first visit is less compared to the other two clusters. Compared to Not Loyal, days since their last visit is lower. Those two patterns are clear indication of new customers. Look at the spent and total visit, it is comparable to Not Loyal but significantly lower than our Loyal customer base.

Cluster	Spent	Recency	Total Visit	Visit Frequency	days_since_first_visit	count
Newcomers	12	164	4	0.0335	213	16373

Not Loyal:

Not Loyal are identified as older customers as their recency and days since their first visit is very high compared to the rest of clusters.

Cluster	Spent	Recency	Total Visit	Visit Frequency	days_since_first_visit	count
Not Loyal	9	563	3	0.0049	576	14811

With these three customer segmentation, Central Perk can effectively develop different targeting strategies based on their profiles to formulate incentives and pricing adjustments.

Final Takeaways

Trend Analysis

1. Weekdays and weekends show a difference in purchasing behavior. Weekends have an average of 2x the transactions to that of weekdays. So we divide our further analysis into weekdays and weekends.

2. Weekdays observe maximum traffic in 8 AM to 11 AM period compared to 9 AM to 12 PM for weekends. Also, traffic trends in restaurant helps in making staffing decisions by allocating more staff at some point of the day, this ensures seamless customer experience and save staffing cost by

optimizing their shifts.

3. Both weekdays and weekends experience a dip in traffic after 4 PM in the evening where discounts can be offered to consumers which will clear inventory faster, get more people in the door, and also lead to an increase in revenue.

4. Drip, Latte, and Cappuccino are the top-selling items and Coffee and food are top-selling categories.

5. Coffees are observed to experience a dip in sales post 12 PM rush hour during weekdays. Discounts can be offered on these products to improve sales.

6. Upon analyzing all individual coffee categories, we observe that Large Drip coffee has a spike in sales at 6 PM on Wednesdays and Thursdays, which is not seen on other days. Central Perk's timings can be further extended as an experiment to analyze if the increase in revenue offsets the operational costs of keeping the store open for an additional hour.

7. Ice combos can be offered to consumers with coffee as a large number of transactions with Small Drip coffee and Small Latte have ice as an add-on. This can be also seen as an opportunity to improve cross-selling as offers and combos can be offered with ice. It is also observed that chai has as extremely low sales among not only all items but also in non-caffeinated drinks.

Cluster Analysis

1. For customers having a Central Perk membership (a customer id) for at least 6 months, we can see that only 25% of customers visited more than 3 times in their lifetime and only 6% of customers repeated their transactions in the last 6 months. This clearly contradicts Central Perk's belief on their high customer loyalty. Thus the team applied the following cluster analysis to generate data-driven consumer segmentation.

2. We observe 3 clusters on analyzing the consumers who have a customer id in the system.

3. By cluster analysis we were able to separate all the customers in three different clusters. Based on their properties, we identified loyal, new and not loyal customers. These customers are different and should be targeted in different ways based on their properties.

Purchase Behavior Analysis

1. Coffee and Food are the most frequently occurring categories. These can either be bundled together or either of them can be marked up to normalize the demand.

2. Almond seems to be the go-to extra which is mostly consumed with Latte and Drip coffee.

Recommendations

1. New customers

Our findings from applying Association rules show that these items are often purchased together -
(Latte + Almonds)
(Cappuccino + Almonds)
(Small Drip + Ice)
(Small Latte + Ice)

The average number of visits from this cluster was four, over the duration of the dataset. These customers can be incentivized for multiple visits by being given a punch card where after a certain number of visits, they get a free beverage/item of their choice.

Beyond this, we can use the list of most purchased items and sell them as a bundle at a lowered price since ice and almonds as add-ons won't affect the overall revenue dramatically.

Multiple visits and lowered prices on most purchased bundles can enable Central Perk to convert these customers from occasional visits to regular to loyal customers which in the long-term can potentially generate much more revenue from them.

2. Non-Loyals

We found these customers' Recency to be 563 days on average. Given the considerable gap between first and last visits, it is possible that they may have come to New York from a different city, state or country.

If they are from a city or state where Central Perk has a location in their vicinity, they can be sent direct mail as part of general coupon mailers or if they chose to share their information, they could be sent email offers asking them to experience Central Perk's other offerings and other limited time promotions being run at Central Perk.

3. Loyals

These customers choose to come to Central Perk for its quality and experience. To not make them feel alienated with general promotions for everyone, Central Perk can enhance their experience further by offering personalized offers and promotions based on their ordering history and exclusive discounts on certain items, combos and seasonal items.

This will also encourage Non-Loyals and New Customers to try Central Perk more often which can convert them into Loyal customers in the future as well.