

# Homework 3: That One That Brings It All Together

*Nakul Agrawal, Samira Arondekar, Sai Akhil Kodali, Xue Ni, Yuting Xin*

*11/2/2019*

## Contents

<b>Introduction</b>	<b>2</b>
The Business Problem . . . . .	2
Who is a loyal customer? . . . . .	2
What does success looks like? . . . . .	2
Approach . . . . .	2
<b>Data Preparation</b>	<b>3</b>
Inspecting and Cleaning Data . . . . .	3
Feature Engineering . . . . .	4
Data Transformation . . . . .	5
<b>Data Analysis</b>	<b>7</b>
Part 1 - Testing hypothesis ‘Is our customer base loyal?’ . . . . .	7
Part 2 - What is the buying behavior (by time) of our customers? . . . . .	12
Part 3 - What is the buying pattern (by type of items) of our customers? . . . . .	19
Part 4 - What are the patterns in demand across categories and time? . . . . .	22
<b>Recommendations</b>	<b>28</b>

## Introduction

Central Perk is a boutique coffee shop which has not yet leveraged the potential of their data to gauge demand and understand their customer base. The client believes that they have a loyal customer base which has remained constant over the years. The owners of the coffee shop and its investors have sought our help in using this data to understand their customer base and generate more revenue from the business.

The data needs to be thoroughly explored to validate the assumptions made by the client and explore dependence of demand on various factors.

## The Business Problem

The goal of the business is to maximize profit. This can be achieved in two ways. First way is by increasing sales to our current / new customers. Second way is to have a constant demand over time by making sure that we sell right products, at right price and at right time.

As sales data has not yet been analysed, it has hampered the business's ability to make informed decisions based on consumer characteristics and demand patterns for the various products. The business wants to understand the variation of volume across seasons, weekdays and hour of day. They also want to check the performance of various items/products and trends for them.

Based on the insights, the business would like recommendations about necessary changes that can be implemented to maximize profit. The business is wary of alienating customers as a result of any changes made.

## Who is a loyal customer?

As the business's current practices assume that they have a loyal customer base, we first check if the customers are indeed loyal. Loyalty is defined as having made at least 5 or more purchases from Central Perk, made at least one purchase in the last 6 months and generated profits of at least \$10. (Note that this is evaluated on sales instances with customer IDs as there is no other way to identify a customer).

## What does success look like?

Success here would result from increasing profit by effectively using sales data to identify patterns in demand and customer behavior and modify strategy/operations accordingly.

## Approach

Our first step is to explore the data by observing repeat visits by customers to gauge loyalty amongst customers. We further explore their contribution towards profits of the business.

Next, we use association rules to understand sales by time element and customer type. This helps us in identifying correlations which can be exploited to increase sales and possible scenarios for low sales.

Further, we dig deeper into item level data using association rules to identify frequently co-occurring combinations which can be leveraged to increase sales and hence profits.

Lastly we observe variation in demand across categories by time to find inconsistent demand. This information can be used to promote sales during low demand periods and optimize staffing to efficiently handle high/low demand to reduce costs.

## Data Preparation

We have a total of about 25 months transaction data spanning over 2016, 2017 and 2018.

*Loading libraries and data:*

```
library(dplyr)
library(lubridate)
library(timeDate)
library(mltools)
library(data.table)
library(tidyr)
library(ggplot2)
library(arules)
library(arulesViz)
library(data.table)

data_2016 <- read.csv("Central Perk Item Sales Summary 2017.csv")
data_2017 <- read.csv("Central Perk Item Sales Summary 2016.csv")
data_2018 <- read.csv("Central Perk Item Sales Summary 2018.csv")

# Merging three datasets together
data <- rbind(data_2016, data_2017, data_2018)
```

## Inspecting and Cleaning Data

On inspection of the dataset, we found the following problems:

- 'Notes' column showed some charges which were accidental and were not exactly a transaction (87 in total). This was reflected clearly in the 'Even.Type'
- Price.Point.Name had 84 missing values.
- Item column had non-interpretable names.
- Discounts, Tax, Gross Sales and Net Sales were not in the right format.

We deal with the above problems below:

```
# Selecting data with Event Type as 'Payment'
data <- filter(data, Event.Type == 'Payment')

# Dropping Notes and Event Type as they do not have any relevant information left
data <- select(data, -c(Notes, Event.Type))

# Dropping row with '' as category as it has no data
data <- data %>%
  filter(Category != '') %>%
  droplevels()

# Dropping rows from Price.Point.Name with missing values (84 rows)
data <- data %>%
  filter(Price.Point.Name != '') %>%
  droplevels()
```

```

# Correcting item name in Item Column
levels(data$Item)[levels(data$Item) == "\U0001f34bLemonade\U0001f34b"] <- "Lemonade"

# Converting Discounts column into correct format
data$Discounts <- trimws(data$Discounts, which = "both")
data$Discounts <- gsub("[$( )]", "", data$Discounts)
data$Discounts <- abs(as.numeric(data$Discounts))

# Correcting Tax, Gross.Sale and Net.Sales
strip_dollars = function(x) { as.numeric(gsub("[\\$, ]", "", x)) }
data[, c("Tax", "Gross.Sales", "Net.Sales")] <-
  lapply(data[, c("Tax", "Gross.Sales", "Net.Sales")] , strip_dollars)

# Converting Date column to right format
data$Date <- as.Date(data$Date,"%m/%d/%y")

```

## Feature Engineering

We performed feature engineering to add additional columns that could give us more information from the existing data.

*Adding following variables:*

1. weekday - Is a Weekday/Weekend (1/0)
2. day\_of\_week - Mon, Tue, Wed, Thu, Fri, Sat
3. profit - (20% of gross sale minus the discount)
4. time\_of\_day - morning (opening to 11 am), afternoon (11am to 4 pm), evening (4 pm to closing)
5. Customer - Is a Registered Customer/Non-Registered Customer (1/0)

```

# Adding a column for weekday
data$Weekday <- isWeekday(data$Date, wday = 1:5)

# Adding a column for day_of_week
data$day_of_week <- as.factor(weekdays(data$Date))

# Adding a column for profit
data$Profit <- (data$Gross.Sales * 0.2) - (data$Discounts)

# Adding a column for time_of_day
data$Time <- hms(data$Time)
data$time_of_day <- ifelse(lubridate::hour(data$Time) <= 10, 'Morning',
                           (ifelse(lubridate::hour(data$Time) <= 15, 'Afternoon',
                                    'Evening'))))

# Adding a flag for customer
data$Customer <- ifelse(is.na(data$Customer.ID), 'Non-Registered Customer',
                        'Registered Customer')
data$Customer <- as.factor(data$Customer)

```

*summary of dataframe*

```
str(data, max.level = 1, strict.width = "wrap")
```

```
## 'data.frame': 221389 obs. of 16 variables:
## $ Date : Date, format: "2016-12-31" "2016-12-31" ...
## $ Time :Formal class 'Period' [package "lubridate"] with 6 slots
## $ Category : Factor w/ 9 levels "Beans","Beers",...: 3 3 3 3 6 3 5 3 3 4
## ...
## $ Item : Factor w/ 69 levels "Lemonade","12oz Columbia",...: 18 29 10 10 23
## 14 17 15 15 8 ...
## $ Qty : int 1 1 2 2 1 1 1 1 2 2 ...
## $ Price.Point.Name: Factor w/ 5 levels "LG","Plain / Choc",...: 1 3 3 3 3 3
## 4 3 3 4 ...
## $ Gross.Sales : num 3.5 3.75 6.5 6.5 4 4 3.75 4 8 1 ...
## $ Discounts : num 0 0 0 0 0 0 0 0 0 0 ...
## $ Net.Sales : num 3.5 3.75 6.5 6.5 4 4 3.75 4 8 1 ...
## $ Tax : num 0 0 0 0 0 0 0 0 0 0 ...
## $ Customer.ID : Factor w/ 31811 levels
## "0000063491abb5e5068c300be93c56c0841250da",...: NA 2913 3755 2484 NA
## 4945 4945 NA 6197 6197 ...
## $ Weekday : logi FALSE FALSE FALSE FALSE FALSE FALSE ...
## $ day_of_week : Factor w/ 7 levels "Friday","Monday",...: 3 3 3 3 3 3 3 3
## 3 ...
## $ Profit : num 0.7 0.75 1.3 1.3 0.8 0.8 0.75 0.8 1.6 0.2 ...
## $ time_of_day : chr "Evening" "Evening" "Evening" "Evening" ...
## $ Customer : Factor w/ 2 levels "Non-Registered Customer",...: 1 2 2 2 1 2
## 2 1 2 2 ...
```

We now have a clean and well prepared dataset which can be used for data transformation, exploration and extracting insights.

## Data Transformation

We transformed data into two levels:

1. First, we grouped by time and got data at an order level. This is done to find insights like what products are frequently bought together, etc.:

```
## 'data.frame': 132955 obs. of 92 variables:
## $ X : int 1 2 3 4 5 6 7 8 9 10 ...
## $ Date : Factor w/ 758 levels "2016-07-15","2016-07-16",...: 1 1 1 1 1 1 1 1
## 1 1 1 ...
## $ Time : Factor w/ 47579 levels "06:40:30","06:52:34",...: 12759 13489
## 13645 14268 14347 14937 15033 15249 15445 15587 ...
## $ Gross.Sales : num 3 6.5 3.25 6.25 3.25 ...
## $ Discounts : num 0 0 0 0 0 0 0 0 0 0 ...
## $ Net.Sales : num 3 6.5 3.25 6.25 3.25 ...
## $ Tax : num 0 0 0 0 0 0 0 0 0 0 ...
## $ Profit : num 0.6 1.3 0.65 1.25 0.65 1.3 1.8 2.25 0.9 3.2 ...
## $ Customer.ID : Factor w/ 31811 levels
## "0000063491abb5e5068c300be93c56c0841250da",...: NA 29867 NA 12921 NA
## 17584 NA 23842 NA 6916 ...
```

```
## $ Weekday : logi TRUE TRUE TRUE TRUE TRUE TRUE ...
## $ day_of_week : Factor w/ 7 levels "Friday","Monday",...: 1 1 1 1 1 1 1 1
##      1 ...
## $ Time_2 : Factor w/ 60 levels "10H 59M 0S","10H 59M 10S",...: 42 3 44 18
##      40 52 29 11 37 4 ...
## $ time_of_day : Factor w/ 3 levels "Afternoon","Evening",...: 3 1 1 1 1 1 1
##      1 1 1 ...
## $ Customer : int 0 1 0 1 0 1 0 1 0 1 ...
## $ Category_Beans : int 0 0 0 0 0 0 0 0 0 0 ...
## $ Category_Beers : int 0 0 0 0 0 0 0 0 0 0 ...
## $ Category_Coffee : int 1 1 1 1 1 1 2 2 1 2 ...
## $ Category_Extras : int 0 1 1 0 0 1 2 0 1 0 ...
## $ Category_Food : int 0 0 0 1 0 0 0 0 0 2 ...
## $ Category_Non_Caffeinated_Drinks: int 0 0 0 0 0 0 0 1 0 1 ...
## $ Category_None : int 0 0 0 0 0 0 0 0 0 0 ...
## $ Category_Tea : int 0 0 0 0 0 0 0 0 0 0 ...
## $ Category_Cereal : int 0 0 0 0 0 0 0 0 0 0 ...
## $ Item_Lemonade : int 0 0 0 0 0 0 0 0 0 0 ...
## $ Item_12oz_Columbia : int 0 0 0 0 0 0 0 0 0 0 ...
## [list output truncated]
```

2. We also grouped data and got it to customer level for deeper consumer analysis:

```
## 'data.frame': 31811 obs. of 87 variables:
## $ X : int 1 2 3 4 5 6 7 8 9 10 ...
## $ Customer.ID : Factor w/ 31811 levels
##      "0000063491abb5e5068c300be93c56c0841250da",...: 1 10 17 18 20 32 35 36
##      43 44 ...
## $ Gross.Sales : num 13.25 2.75 4.25 5 3.5 ...
## $ Discounts : num 0 0 0 0 0 0 0 0 0 0 ...
## $ Net.Sales : num 13.25 2.75 4.25 5 3.5 ...
## $ Tax : num 0 0 0 0 0 0 0 0 0 0 ...
## $ Profit : num 2.65 0.55 0.85 1 0.7 1.25 1.9 0.6 1.35 0.7 ...
## $ days_since_last_order : int 550 754 677 684 716 603 609 664 710 669 ...
## $ total_orders : int 2 1 1 1 1 1 1 1 1 1 ...
## $ Category_Beans : int 0 0 0 0 0 0 0 0 0 0 ...
## $ Category_Beers : int 0 0 0 0 0 0 0 0 0 0 ...
## $ Category_Coffee : int 2 1 1 1 1 1 2 1 1 1 ...
## $ Category_Extras : int 0 0 0 0 1 0 0 0 0 0 ...
## $ Category_Food : int 1 0 0 0 0 0 0 0 1 0 ...
## $ Category_Non_Caffeinated_Drinks: int 1 0 0 0 0 1 0 0 0 0 ...
## $ Category_None : int 0 0 0 0 0 0 0 0 0 0 ...
## $ Category_Tea : int 0 0 0 0 0 0 0 0 0 0 ...
## $ Category_Cereal : int 0 0 0 0 0 0 0 0 0 0 ...
## $ Item_Lemonade : int 0 0 0 0 0 0 0 0 0 0 ...
## $ Item_12oz_Columbia : int 0 0 0 0 0 0 0 0 0 0 ...
## [list output truncated]
```

Our data transformation is now complete and we proceed with our data analysis.

# Data Analysis

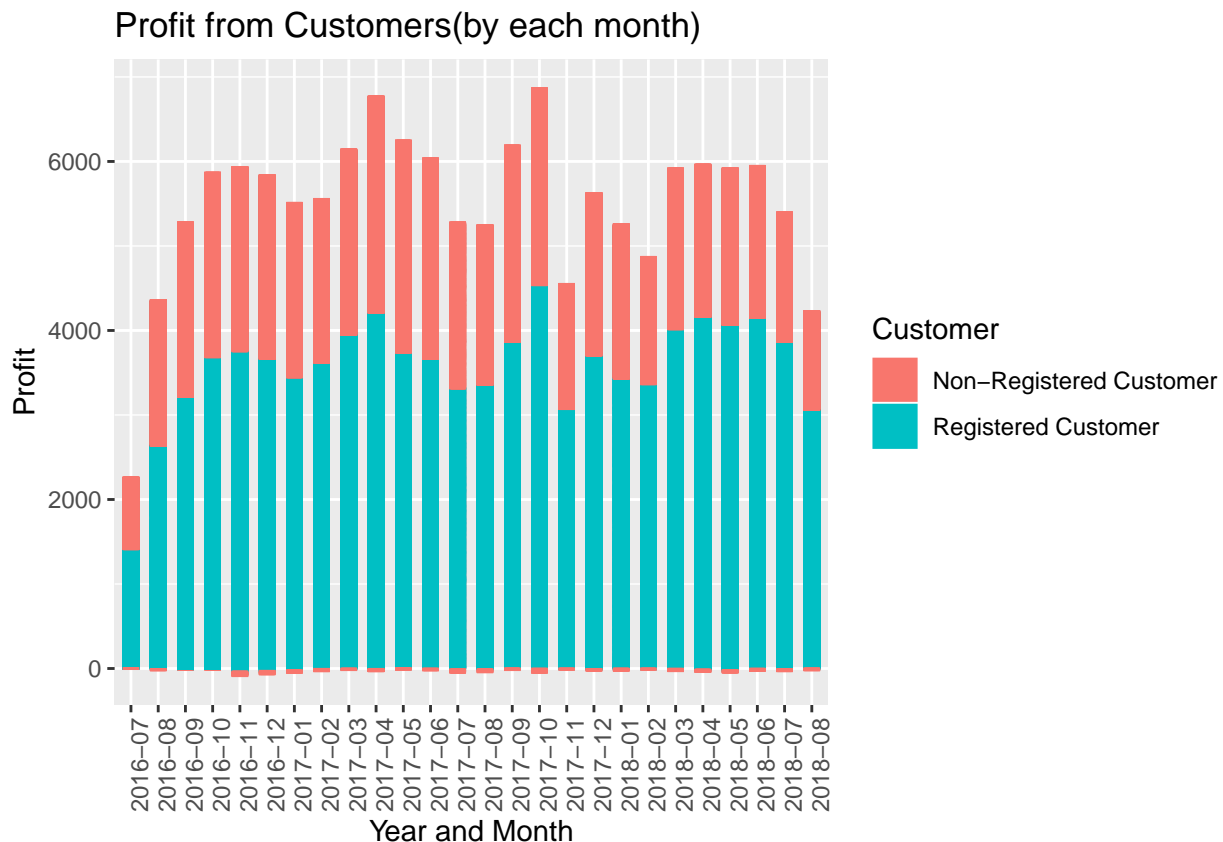
## Part 1 - Testing hypothesis ‘Is our customer base loyal?’

### Overview

We want to analyze the hypothesis that most of the profits are brought in by our loyal customers. From the dataset we can see some customers have an ID, and we assume they have registered an account with us similar to the ‘Starbucks Rewards Program’. For customers with customer ID, we identify them as “Registered Customers”. For those customers without any ID, we consider them as “Non-Registered Customers”.

### Analysis of Profit by Customer type

```
ggplot(data, aes(x = format(as.Date(data$Date), "%Y-%m"), y = Profit,
                  color = Customer, fill = Customer, width = 0.5)) +
  geom_bar(stat = "identity") +
  labs(title = 'Profit from Customers(by each month)',
       y = "Profit", x = "Year and Month") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

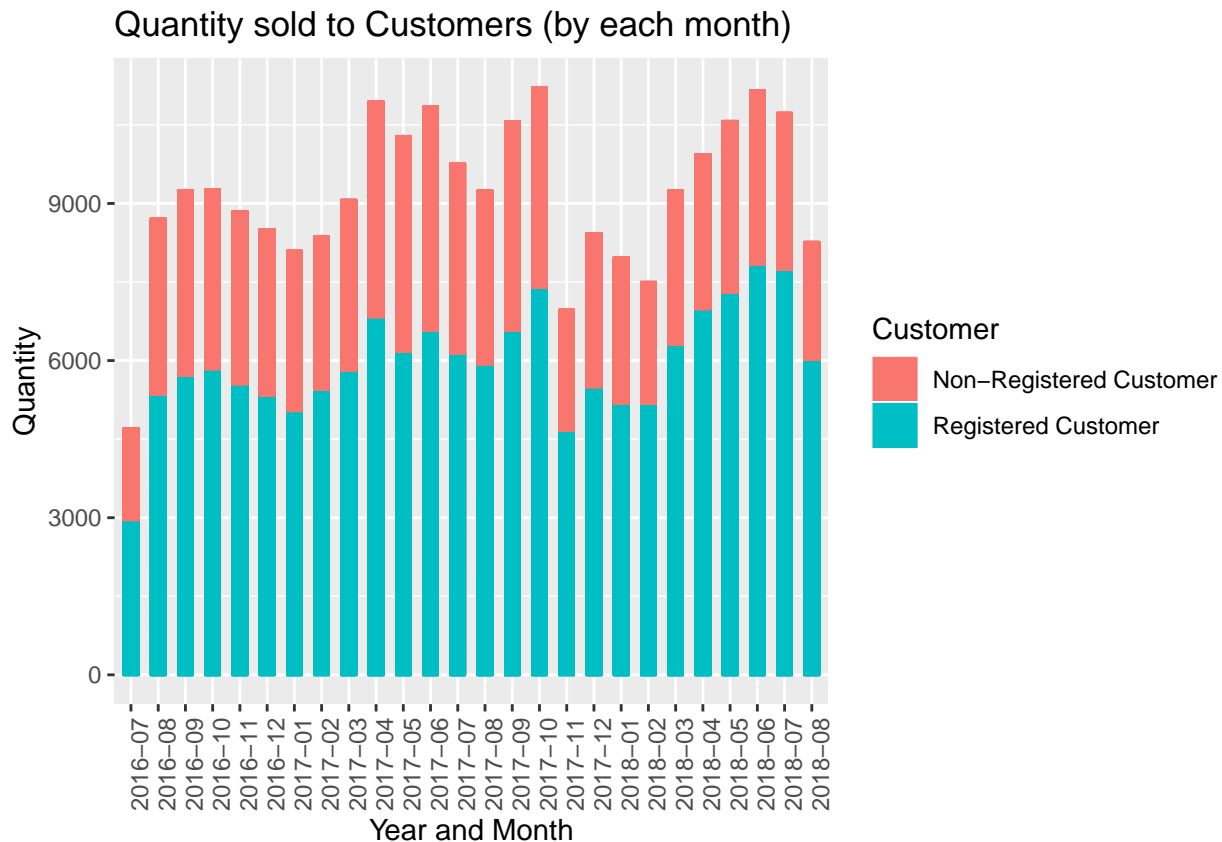


### Interpretation

As we can see above, during the three year period, the profit distribution by Registered vs Non-Registered Customers is relatively stable. By calculation, we see that about two thirds of profits are brought by registered customers.

### Analysis of Quantity sold by Customer type

```
ggplot(data, aes(x = format(as.Date(data$Date), "%Y-%m"), y = Qty,
                    color = Customer, fill = Customer, width = 0.5)) +
  geom_bar(stat = "identity") +
  labs(title = 'Quantity sold to Customers (by each month)',
        y = "Quantity", x = "Year and Month") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



### Interpretation

As we can see above, the analysis of quantity sold matches our result from previous graph. By calculation, we see that again, about two thirds of quantity are sold to registered customers. Based on analysis above, we can see the reason why our client believes the priority is to maintain loyalty customers. However, are these registered customers really loyal to our coffee shop? We study the data further in this direction.

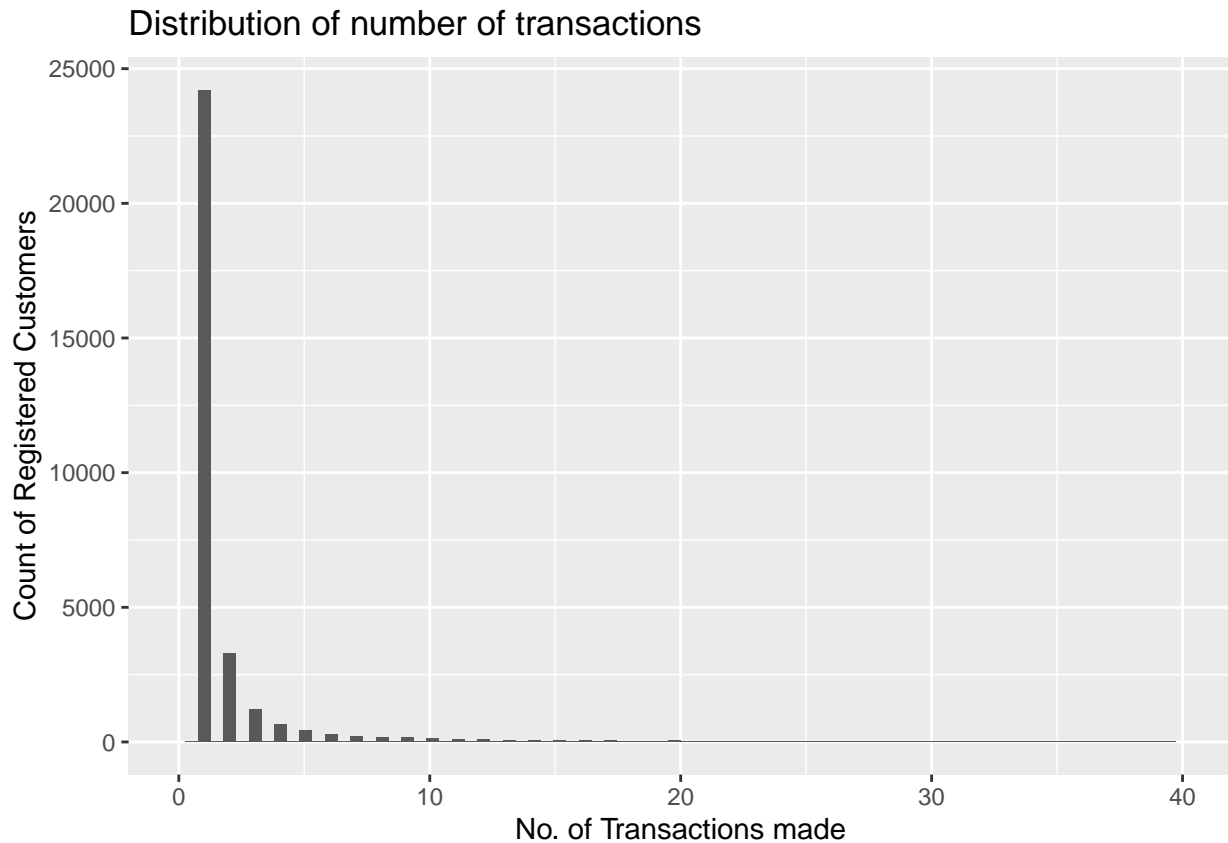
### Overview

We further explore our registered customers. We want to know for how many days they have been our customers, how many transactions they have made, along with how much money they have spent at our coffee shop.

### Analysis of Transactions made

```
ggplot(data_customers, aes(x = total_orders)) +
  geom_histogram(bins = 80) +
  labs(title = 'Distribution of number of transactions',
        y = "Count of Registered Customers",
        x = "No. of Transactions made") +
  xlim(0,40)
```





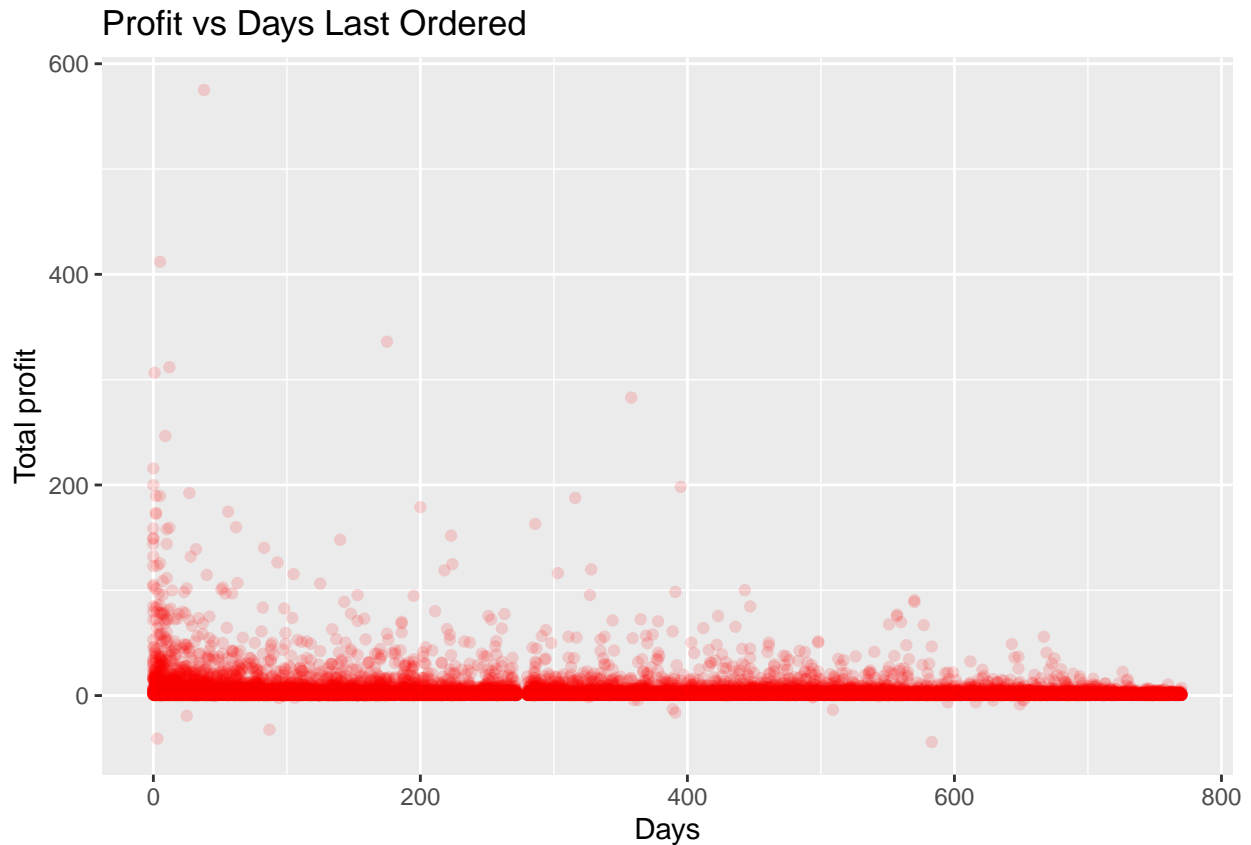
#### *Interpretation*

We can see from the above graph that majority of our registered customers make very few transactions with us. Only a handful of customers make over 20 purchases from us. Of all the registered customers, 76% made just a single transaction.

Next we analyze the timeline of transactions of our customers.

#### *Analysis of profit vs last ordered date*

```
ggplot(data_customers, aes(x = days_since_last_order, y = Profit)) +  
  geom_point(alpha = 0.15, color = 'red') +  
  labs(title = 'Profit vs Days Last Ordered', y = "Total profit", x = "Days")
```



#### Interpretation

In this plot, each point represents a customer. We see that most of our customers have not made a purchase with us for quite some time. For some reason, our customers do not return to our store to make another purchase again.

#### Recency and Frequency Analysis for Loyal Customers

Finally we look at customers with whom we have had a good relation. We have set our expectations low and defined a loyal customer as below: 1. The customer should have visited the cafe at least 5 times 2. The customer should have visited at least once in last 6 months 3. The customer should have at least made a profit of \$10 for the business

```
data_customers %>% filter(days_since_last_order < 180 & total_orders > 4 & Profit > 10) %>%
  summarise(percentage_of_people = (n()/31811)*100, Avg_Profit = mean(Profit))
```

```
##   percentage_of_people Avg_Profit
## 1             2.307378   34.68767
```

#### Interpretation

Only 2.3% of our total 'Registered Customers' actually pass the above criteria and are safe to be said as loyal. This is a very low number as compared to the total number of registered customers. These customers generate a profit of \$34 on average.

#### Analysing Extras - Loyal Customers Vs All

We found that loyal customers are more likely to buy 'extras' with their coffee.

```

# For Loyal Customers
data_customers %>%
  filter(days_since_last_order < 180 & total_orders > 4 &
         Profit > 10 & Category_Extras > 1) %>%
  summarise(percentage_of_people = (n()/734)*100, Avg_Profit = mean(Profit))

##   percentage_of_people Avg_Profit
## 1             77.11172     36.941

# For all Registered Customers
data_customers %>% filter(Category_Extras > 1) %>%
  summarise(percentage_of_people = (n()/31811)*100, Avg_Profit = mean(Profit))

##   percentage_of_people Avg_Profit
## 1             14.64588     9.213995

```

### *Interpretation*

Only 15% of all our customers have ever bought an ‘extra’. Whereas, out of the loyal customers, around 80% have bought extras.

Hence, it is safe to say that buying extras is an important characteristic of a loyal customer and if the cafe can start giving extras at a low price or for free, they can make more customers as their loyal customers. We will deep dive into Extras in the next part of our analysis.

### *Conclusion of Part 1*

From our analysis on our customers, we can see that most of our profits are brought in by registered customers and similarly, we see most of our products are sold to these existing customers.

However, most of our registered customers make only few transactions with us and they are not loyal to us in terms of duration and frequency. From these evidences we can conclude that our client’s hypothesis that their customer base is loyal, is incorrect. Even though our client does a good job making customers register an account with them, client fails to turn them into loyal customers.

We can see that most of the customers are not returning. Even the customers who have made us good profit in the past, have stopped coming to our shop.

We also saw that buying products from category Extras is an important characteristic of a loyal customer. Hence, our client needs to increase loyal customers from their registered customer base.

## Part 2 - What is the buying behavior (by time) of our customers?

### Overview

From previous observations, we see that our client's business is very easily affected by time variables (such as season and weekday). Hence, we will run association rules here to see how each category of product is affected by time and different types of customers.

Following is how we have separated our time variables:

- By Season
  - Spring: March-May
  - Summer: June-August
  - Fall: September-November
  - Winter: December-February
- By Time of Day
  - Morning: Before 10am
  - Afternoon: 10am-3pm
  - Evening: After 3pm
- By Day of Week
  - Weekday: Monday-Friday
  - Weekend: Saturday & Sunday

```
transactions <- read.transactions("arules_1.csv", format = "basket", sep = ',')
```

```
rules <- apriori(transactions, parameter = list(supp = 0.0001, conf = 0.05))
```

1. *Analysis of rules with largest and smallest lift* First, we want to take a look at all rules and then find rules with the largest lift and smallest lift.

```
rules <- sort(rules, by = "lift", decreasing = TRUE)
rules %>%
  subset(subset = !(rhs %pin% "Winter" | rhs %pin% "Fall" |
    rhs %pin% "Summer" | rhs %pin% "Spring" |
    rhs %pin% "Morning" | rhs %pin% "Evening" |
    rhs %pin% "Afternoon" | rhs %pin% "weekend" |
    rhs %pin% "weekday" | rhs %pin% "Non-Registered Customer" |
    rhs %pin% "Registered Customer" | rhs %pin% "Extras" )) %>%
  subset(subset = lift > 2.3) %>%
  inspect()
```

##	lhs	rhs	support	confidence	lift	count
## [1]	{Evening,					
##	Non-Registered Customer,					
##	weekend,					
##	Winter}	=> {Non-Caffeinated Drinks}	0.0002032612	0.06122449	2.794740	45
## [2]	{Evening,					
##	weekend,					
##	Winter}	=> {Non-Caffeinated Drinks}	0.0005962329	0.05916629	2.700789	132
## [3]	{Evening,					
##	Registered Customer,					

```

##      weekday,
##      Winter}                => {Non-Caffeinated Drinks} 0.0005555806 0.05885167 2.686427 123
## [4] {Evening,
##      Registered Customer,
##      Winter}                => {Non-Caffeinated Drinks} 0.0009485523 0.05856107 2.673162 210
## [5] {Evening,
##      Registered Customer,
##      weekend,
##      Winter}                => {Non-Caffeinated Drinks} 0.0003929717 0.05815508 2.654630 87
## [6] {Evening,
##      Winter}                => {Non-Caffeinated Drinks} 0.0015312345 0.05760408 2.629478 339
## [7] {Evening,
##      weekday,
##      Winter}                => {Non-Caffeinated Drinks} 0.0009350016 0.05665025 2.585938 207
## [8] {Evening,
##      Non-Registered Customer,
##      Winter}                => {Non-Caffeinated Drinks} 0.0005826821 0.05611135 2.561339 129
## [9] {Evening,
##      Non-Registered Customer,
##      weekday,
##      Winter}                => {Non-Caffeinated Drinks} 0.0003794209 0.05370844 2.451652 84
## [10] {Evening,
##      Non-Registered Customer,
##      weekday,
##      Winter}                => {Tea} 0.0008311125 0.11764706 2.423549 184
## [11] {Evening,
##      Non-Registered Customer,
##      Winter}                => {Tea} 0.0011924658 0.11483254 2.365569 264
## [12] {Evening,
##      weekday,
##      Winter}                => {Tea} 0.0018654862 0.11302682 2.328371 413

```

```

rules <- sort(rules,by = "lift", decreasing = FALSE)
rules %>%
  subset(subset = !(rhs %pin% "Winter" | rhs %pin% "Fall" | rhs %pin% "Summer" |
    rhs %pin% "Spring" | rhs %pin% "Morning" |
    rhs %pin% "Evening" | rhs %pin% "Afternoon" |
    rhs %pin% "weekend" | rhs %pin% "weekday" |
    rhs %pin% "Non-Registered Customer" |
    rhs %pin% "Registered Customer" | rhs %pin% "Extras" )) %>%
  subset(subset = lift < 0.7) %>%
  inspect()

```

```

##      lhs                rhs                support confidence    lift count
## [1] {Evening,
##      Non-Registered Customer,
##      Summer,
##      weekend}            => {Food} 0.0004245901 0.07448494 0.5922786    94
## [2] {Afternoon,
##      Non-Registered Customer,
##      Summer,
##      weekend}            => {Food} 0.0013505578 0.08356624 0.6644900    299
## [3] {Evening,
##      Non-Registered Customer,

```

```
##      Summer}                                => {Food} 0.0014589638 0.08365708 0.6652123   323
## [4] {Afternoon,
##      Non-Registered Customer,
##      Summer}                                => {Food} 0.0036270834 0.08408377 0.6686052   803
## [5] {Afternoon,
##      Non-Registered Customer,
##      Summer,
##      weekday}                              => {Food} 0.0022765256 0.08439384 0.6710707   504
```

### Interpretation

We want to see relationships with the largest and weakest lift. As we can see, in winter evening, customers are very likely to buy tea and non-caffeinated drinks, with a lift over 2.3, which is very high. Then we look into relationships with the weakest lift. This tells us about the unlikeliness of a rule. Top 10 rules all related to food. We can conclude from the results that customers are very less likely to buy food on a summer afternoon and summer evening. It is very clear that our client can have some promotions here, relating to tea and non-caffeinated drinks in winter to boost sales. Also, client should consider cutting down the order for food from suppliers in summer since there is not much sales in food in this season.

### 2. Analysis of rules related to just coffee and food

Since most of our profits are brought in by food category and coffee category, we want to focus on rules relating to coffee and food.

#### a. coffee

```
rules <- sort(rules, by = "lift", decreasing = TRUE)
rules %>%
  subset(subset = !(rhs %pin% "Winter" | rhs %pin% "Fall" | rhs %pin% "Summer" |
    rhs %pin% "Spring" | rhs %pin% "Morning" | rhs %pin% "Evening" |
    rhs %pin% "Afternoon" | rhs %pin% "weekend" |
    rhs %pin% "weekday" | rhs %pin% "Non-Registered Customer" |
    rhs %pin% "Registered Customer" | rhs %pin% "Extras" )) %>%
  subset(subset = lift > 1.12) %>%
  subset(subset = (rhs %pin% 'Coffee')) %>%
  inspect()
```

```
##      lhs                                rhs      support confidence    lift count
## [1] {Morning,
##      Non-Registered Customer,
##      weekday,
##      Winter}                              => {Coffee} 0.013695289 0.6847335 1.169736 3032
## [2] {Morning,
##      Non-Registered Customer,
##      Winter}                              => {Coffee} 0.019454357 0.6811640 1.163639 4307
## [3] {Morning,
##      Non-Registered Customer,
##      weekend,
##      Winter}                              => {Coffee} 0.005759068 0.6728232 1.149390 1275
## [4] {Morning,
##      Non-Registered Customer,
##      Spring,
##      weekend}                              => {Coffee} 0.007290302 0.6685998 1.142175 1614
## [5] {Morning,
##      Non-Registered Customer,
```

```

##      Spring}                                => {Coffee} 0.023632504  0.6639594 1.134248  5232
## [6] {Morning,
##      Non-Registered Customer,
##      Spring,
##      weekday}                                => {Coffee} 0.016342202  0.6619100 1.130747  3618
## [7] {Fall,
##      Morning,
##      Non-Registered Customer,
##      weekday}                                => {Coffee} 0.016667419  0.6608166 1.128879  3690
## [8] {Afternoon,
##      Non-Registered Customer,
##      weekend,
##      Winter}                                => {Coffee} 0.008410497  0.6588818 1.125574  1862
## [9] {Fall,
##      Morning,
##      Non-Registered Customer} => {Coffee} 0.023727359  0.6581882 1.124389  5253
## [10] {Morning,
##      weekday,
##      Winter}                                => {Coffee} 0.037896924  0.6567001 1.121847  8390

```

```

rules <- sort(rules, by = "lift", decreasing = FALSE)
rules %>%
  subset(subset = !(rhs %pin% "Winter" | rhs %pin% "Fall" | rhs %pin% "Summer" |
    rhs %pin% "Spring" | rhs %pin% "Morning" | rhs %pin% "Evening" |
    rhs %pin% "Afternoon" | rhs %pin% "weekend" |
    rhs %pin% "weekday" | rhs %pin% "Non-Registered Customer" |
    rhs %pin% "Registered Customer" | rhs %pin% "Extras" )) %>%
  subset(subset = lift < 0.85) %>%
  subset(subset = (rhs %pin% 'Coffee')) %>%
  inspect()

```

```

##      lhs                                rhs      support confidence      lift count
## [1] {Evening,
##      Registered Customer,
##      Summer,
##      weekday}                                => {Coffee} 0.007592936  0.4605479 0.7867582  1681
## [2] {Evening,
##      Registered Customer,
##      Summer}                                => {Coffee} 0.012710601  0.4665893 0.7970786  2814
## [3] {Evening,
##      Registered Customer,
##      Summer,
##      weekend}                                => {Coffee} 0.005117666  0.4758505 0.8128996  1133
## [4] {Evening,
##      Summer,
##      weekday}                                => {Coffee} 0.013550748  0.4800768 0.8201195  3000
## [5] {Evening,
##      Summer}                                => {Coffee} 0.021613442  0.4837242 0.8263504  4785
## [6] {Evening,
##      Summer,
##      weekend}                                => {Coffee} 0.008062695  0.4899808 0.8370385  1785
## [7] {Afternoon,
##      Registered Customer,
##      Summer,

```

```
##      weekday}                => {Coffee} 0.022814942  0.4968034 0.8486936  5051
## [8] {Afternoon,
##      Registered Customer,
##      Summer}                => {Coffee} 0.041393017  0.4970170 0.8490586  9164
## [9] {Afternoon,
##      Registered Customer,
##      Summer,
##      weekend}                => {Coffee} 0.018578075  0.4972797 0.8495073  4113
```

### Interpretation

Here we generate rules relating to coffee with lift greater than 1.1 and lift less than 0.85. # why??

We can see that new customers are more likely to buy coffee and more quantity is sold in the morning period. However, we should keep in mind that the lift is about 1.1, which is not very significant. Registered Customers are less likely to buy coffee in summer and in the afternoon or evening.

We can conclude here that morning is the period that our client sells most coffee. Also, non-registered customers are more likely to buy coffee from our client in the morning. Our client should definitely come out with some strategies to attract customers in the morning period. We can also conclude that the sales of our coffee are relatively stable since there are not very large or very small lift from these rules.

### b. food

Now we explore the food section in the store. This will help us understand how the food items fare with customers.

```
rules <- sort(rules, by = "lift", decreasing = TRUE)
rules %>%
  subset(subset = !(rhs %pin% "Winter" | rhs %pin% "Fall" | rhs %pin% "Summer" |
    rhs %pin% "Spring" | rhs %pin% "Morning" | rhs %pin% "Evening" |
    rhs %pin% "Afternoon" | rhs %pin% "weekend" |
    rhs %pin% "weekday" | rhs %pin% "Non-Registered Customer" |
    rhs %pin% "Registered Customer" | rhs %pin% "Extras" )) %>%
  subset(subset = lift > 1.3) %>%
  subset(subset = (rhs %pin% 'Food')) %>%
  inspect()
```

```
##      lhs                rhs                support confidence    lift count
## [1] {Morning,
##      Registered Customer,
##      weekend,
##      Winter}          => {Food} 0.003044401  0.1872222 1.488727    674
## [2] {Morning,
##      Registered Customer,
##      Spring,
##      weekend}          => {Food} 0.003636117  0.1800492 1.431689    805
## [3] {Fall,
##      Morning,
##      Registered Customer,
##      weekend}          => {Food} 0.003292832  0.1791155 1.424265    729
## [4] {Morning,
##      weekend,
##      Winter}          => {Food} 0.004349790  0.1752502 1.393529    963
## [5] {Morning,
##      Registered Customer,
```



```
##      Winter}                => {Food} 0.009359050  0.1734181 1.378961  2072
## [6] {Morning,
##      Registered Customer,
##      weekend}                => {Food} 0.013636569  0.1721798 1.369114  3019
## [7] {Morning,
##      Spring,
##      weekend}                => {Food} 0.005307376  0.1706609 1.357036  1175
## [8] {Fall,
##      Morning,
##      weekend}                => {Food} 0.004918921  0.1683934 1.339006  1089
## [9] {Morning,
##      Registered Customer,
##      weekday,
##      Winter}                => {Food} 0.006314648  0.1674653 1.331626  1398
```

```
rules <- sort(rules, by = "lift", decreasing = FALSE)
rules %>%
  subset(subset = !(rhs %pin% "Winter" | rhs %pin% "Fall" | rhs %pin% "Summer" |
    rhs %pin% "Spring" | rhs %pin% "Morning" | rhs %pin% "Evening" |
    rhs %pin% "Afternoon" | rhs %pin% "weekend" |
    rhs %pin% "weekday" | rhs %pin% "Non-Registered Customer" |
    rhs %pin% "Registered Customer" | rhs %pin% "Extras" )) %>%
  subset(subset = lift < 0.75) %>%
  subset(subset = (rhs %pin% 'Food')) %>%
  inspect()
```

	lhs	rhs	support	confidence	lift	count
## [1]	{Evening, Non-Registered Customer, Summer, weekend}	=> {Food}	0.0004245901	0.07448494	0.5922786	94
## [2]	{Afternoon, Non-Registered Customer, Summer, weekend}	=> {Food}	0.0013505578	0.08356624	0.6644900	299
## [3]	{Evening, Non-Registered Customer, Summer}	=> {Food}	0.0014589638	0.08365708	0.6652123	323
## [4]	{Afternoon, Non-Registered Customer, Summer}	=> {Food}	0.0036270834	0.08408377	0.6686052	803
## [5]	{Afternoon, Non-Registered Customer, Summer, weekday}	=> {Food}	0.0022765256	0.08439384	0.6710707	504
## [6]	{Evening, Fall, Non-Registered Customer, weekend}	=> {Food}	0.0003749040	0.08811040	0.7006236	83
## [7]	{Evening, Non-Registered Customer, Summer, weekday}	=> {Food}	0.0010343737	0.08811081	0.7006268	229
## [8]	{Afternoon,					

```

##      Summer,
##      weekday}          => {Food} 0.0064682235 0.08872917 0.7055438 1432
## [9] {Evening,
##      Summer,
##      weekend}           => {Food} 0.0014725146 0.08948669 0.7115673 326
## [10] {Evening,
##       Summer}          => {Food} 0.0040697412 0.09108370 0.7242663 901
## [11] {Afternoon,
##       Registered Customer,
##       Summer,
##       weekday}         => {Food} 0.0041916979 0.09127570 0.7257929 928
## [12] {Afternoon,
##       Summer}          => {Food} 0.0115407200 0.09128912 0.7258997 2555
## [13] {Evening,
##       Summer,
##       weekday}         => {Food} 0.0025972266 0.09201472 0.7316694 575

```

### *Interpretation*

We can clearly see some trends here. Registered customers are more likely to buy food in morning and generally customers are less likely to buy food in afternoon or evening. Non-registered customers are less likely to try our food and registered customers are more likely to buy food. This means the quality of our food is approved by our registered customers and our client should find a way to make more customers try our food.

### *Conclusions of Part 2:*

There are many rules that can be generated. Here we only list that we think are most applicable and have a huge potential. Later in the report, we have provided our recommendations based on these insights.

1. Customers are much more likely to buy tea and non-caffenaited drinks in winter evening.
2. Most of the coffee and food are sold in the morning. Sales of them drop a lot in afternoon and evening, especially in the summer.
3. Registered customers like our food and non-registered customers are less likely to buy our food. This indicates that our food has good taste but not attractive to new customers.

## Part 3 - What is the buying pattern (by type of items) of our customers?

### Overview

Sometimes customers buy more than one item in one transaction. Here we want to know which set of items are more or less likely to appear together.

We took the following steps for convenience of our analysis and improve the insights: 1. Combined large/small orders of latte, drip, and tea together 2. Analyze items which sold over 1,000 times.

```
itemarules = read.transactions("items_1.csv", format = "basket", sep = ",")
```

```
## Warning in asMethod(object): removing duplicated items in transactions
```

```
rules <- apriori(itemarules, parameter = list(supp = 0.0005, conf = 0.05))
```

```
rules <- sort(rules, by = "lift", decreasing = TRUE)
```

```
rules %>%  
  subset(subset = lift > 3) %>%  
  inspect()
```

##	lhs	rhs	support	confidence
## [1]	{Cappucino,Lenka Bar}	=> {Soy}	0.0005039261	0.10000000
## [2]	{Extra Shot}	=> {Chai}	0.0013087036	0.06538895
## [3]	{Chai}	=> {Extra Shot}	0.0013087036	0.10363311
## [4]	{Soy}	=> {Chai}	0.0010228948	0.05499393
## [5]	{Chai}	=> {Soy}	0.0010228948	0.08100060
## [6]	{Ice,Soy}	=> {Latte}	0.0030085141	0.84566596
## [7]	{Almond,Ice}	=> {Latte}	0.0083335840	0.81771218
## [8]	{Cappucino,Latte}	=> {Soy}	0.0005640964	0.06762849
## [9]	{Lenka Bar,Soy}	=> {Cappucino}	0.0005039261	0.48905109
## [10]	{Almond,Croissant}	=> {Latte}	0.0018276723	0.69034091
## [11]	{Croissant,Soy}	=> {Latte}	0.0007370860	0.69014085
## [12]	{Donut,Soy}	=> {Latte}	0.0006844370	0.68421053
## [13]	{Latte,Lenka Bar}	=> {Almond}	0.0012109269	0.19143876
## [14]	{Almond,Drip,Ice}	=> {Latte}	0.0011733205	0.68122271
## [15]	{Drip,Latte}	=> {Almond}	0.0025196306	0.18326039
## [16]	{Cappucino,Soy}	=> {Lenka Bar}	0.0005039261	0.11921708
## [17]	{Drip,Ice,Latte}	=> {Almond}	0.0011733205	0.18076477
## [18]	{Almond,Donut}	=> {Latte}	0.0018502362	0.64229765
## [19]	{Latte}	=> {Soy}	0.0119438010	0.06037564
## [20]	{Soy}	=> {Latte}	0.0119438010	0.64213506
## [21]	{Latte}	=> {Almond}	0.0351920936	0.17789522
## [22]	{Almond}	=> {Latte}	0.0351920936	0.63426867
## [23]	{Almond,Lenka Bar}	=> {Latte}	0.0012109269	0.61923077
## [24]	{Ice,Latte}	=> {Soy}	0.0030085141	0.05758710
## [25]	{Almond,Drip}	=> {Latte}	0.0025196306	0.60469314
## [26]	{Cappucino,Latte}	=> {Almond}	0.0014140016	0.16952209
## [27]	{Chai}	=> {Almond}	0.0021134812	0.16736152
##	lift	count		
## [1]	5.376304	67		
## [2]	5.177995	174		
## [3]	5.177995	174		

```
## [4] 4.354838 136
## [5] 4.354838 136
## [6] 4.274822 400
## [7] 4.133516 1108
## [8] 3.635913 75
## [9] 3.557991 67
## [10] 3.489657 243
## [11] 3.488646 98
## [12] 3.458668 91
## [13] 3.450309 161
## [14] 3.443565 156
## [15] 3.302910 335
## [16] 3.302214 67
## [17] 3.257932 156
## [18] 3.246800 246
## [19] 3.245978 1588
## [20] 3.245978 1588
## [21] 3.206213 4679
## [22] 3.206213 4679
## [23] 3.130197 161
## [24] 3.096058 400
## [25] 3.056710 335
## [26] 3.055304 188
## [27] 3.016364 281
```

### Interpretation

Here we only select rules with a lift more than 3. Several observations we can draw from our list of rules.

1. Customers are more likely to buy more items such as almond milk, croissant, and donut when they purchase Latte. It means that increasing the sales of Latte will also increase the sales of many other products.
2. Customers have a very high lift to order extra shot when they order Chai tea. It would be a great idea to introduce a new tea drink with a stronger flavor.

```
rules <- sort(rules, by = "lift", decreasing = FALSE)
rules %>%
  subset(subset = lift < 0.3) %>%
  inspect()
```

##	lhs	rhs	support	confidence	lift
## [1]	{Cappucino}	=> {Drip}	0.0078672644	0.05723666	0.1477461
## [2]	{Latte,Soy}	=> {Drip}	0.0007070008	0.05919395	0.1527985
## [3]	{Cappucino,Croissant}	=> {Drip}	0.0007596498	0.06257745	0.1615324
## [4]	{Americano}	=> {Drip}	0.0052423358	0.06402131	0.1652594
## [5]	{Soy}	=> {Drip}	0.0011958844	0.06429438	0.1659643
## [6]	{Almond,Cappucino}	=> {Drip}	0.0007746924	0.06535533	0.1687030
## [7]	{Latte}	=> {Drip}	0.0137489094	0.06950042	0.1794028
## [8]	{Almond,Latte}	=> {Drip}	0.0025196306	0.07159649	0.1848134
## [9]	{Croissant,Latte}	=> {Drip}	0.0010755438	0.07175113	0.1852126
## [10]	{Almond}	=> {Drip}	0.0041667920	0.07509828	0.1938526
## [11]	{Cappucino,Donut}	=> {Drip}	0.0006393092	0.07805326	0.2014804
## [12]	{Mocha}	=> {Drip}	0.0015343422	0.08245756	0.2128493

```

## [13] {Tea}          => {Drip}  0.0056033575 0.08300836 0.2142711
## [14] {Chai}         => {Drip}  0.0010605012 0.08397856 0.2167755
## [15] {Donut,Latte}   => {Drip}  0.0011357141 0.08982748 0.2318734
## [16] {Hot Chocalate} => {Drip}  0.0012259695 0.09101061 0.2349274
## [17] {Espresso}      => {Drip}  0.0041216643 0.09932935 0.2564007
## [18] {Macchiato}     => {Drip}  0.0021210024 0.09992913 0.2579490
## [19] {Donut,Drip}    => {Latte} 0.0011357141 0.05322524 0.2690523
## [20] {Almond,Cappucino} => {Ice}   0.0007446072 0.06281726 0.2750331
##      count
## [1] 1046
## [2]   94
## [3]  101
## [4]  697
## [5]  159
## [6]  103
## [7] 1828
## [8]  335
## [9]  143
## [10] 554
## [11]   85
## [12] 204
## [13] 745
## [14] 141
## [15] 151
## [16] 163
## [17] 548
## [18] 282
## [19] 151
## [20]  99

```

*Interpretation* Here we only select rule with a lift less than 0.3. It is very surprising that 19 out of 20 rules we get are relating to Drip. A conclusion here in most of the time customers who buy Drip are not likely to buy anything else with it. And hence Drip is a product which gets sold without pairing with any other product. This also shows that customers who purchase Drip are very different from customers who purchase other products.

## Part 4 - What are the patterns in demand across categories and time?

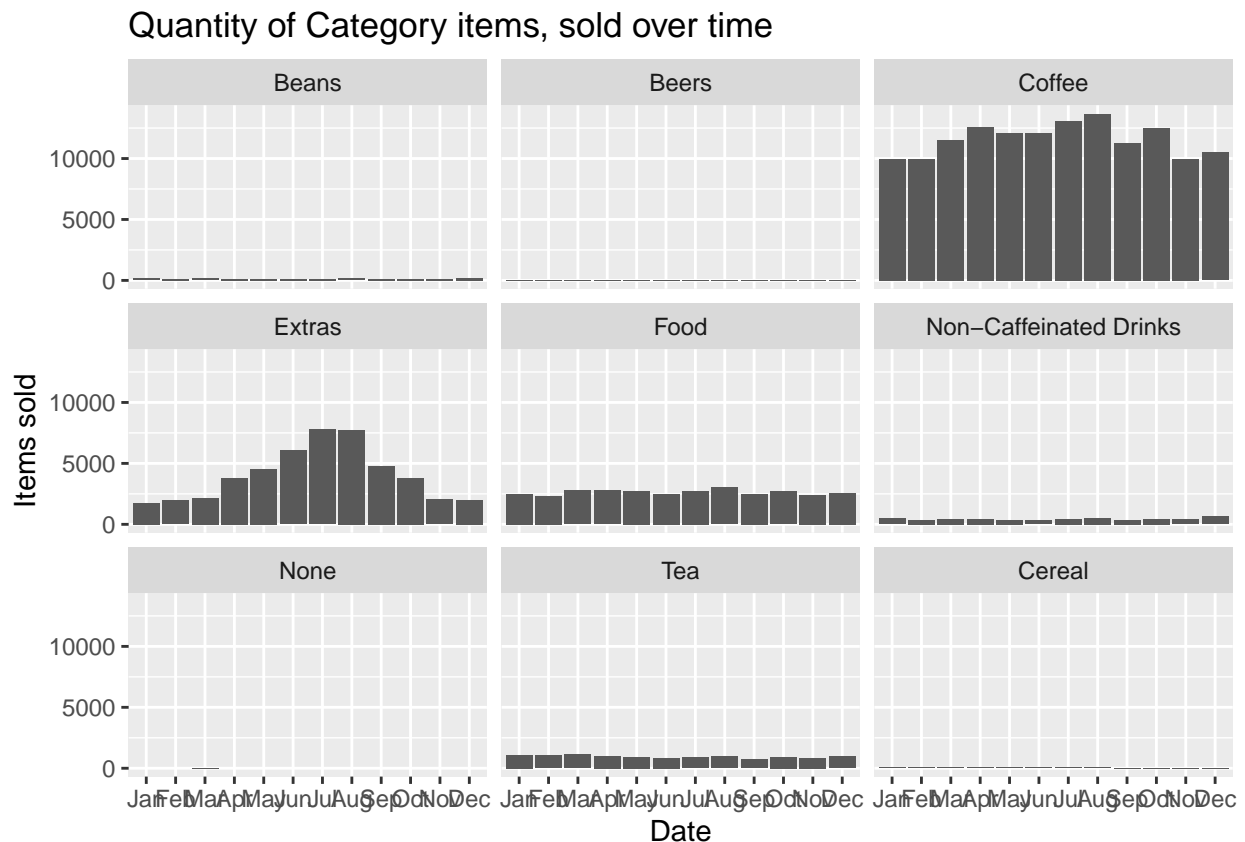
### Overview

The main objective of this part is to find out what are coffee purchasing patterns regarding to sales volume, across categories and items. That way we can find the potential inconsistent demand from customers that can lead us to recommendations.

### Sales Volume Analysis by Category

First, let's look at sales by Category

```
data %>%
  mutate(month = lubridate::month(Date, label = TRUE)) %>%
  group_by(month, Category) %>%
  summarise(tol_qty = sum(Qty)) %>%
  ggplot(aes(month, tol_qty)) +
  geom_bar(stat = 'identity') +
  labs(title = 'Quantity of Category items, sold over time', y = "Items sold", x = "Date") +
  facet_wrap(~Category)
```



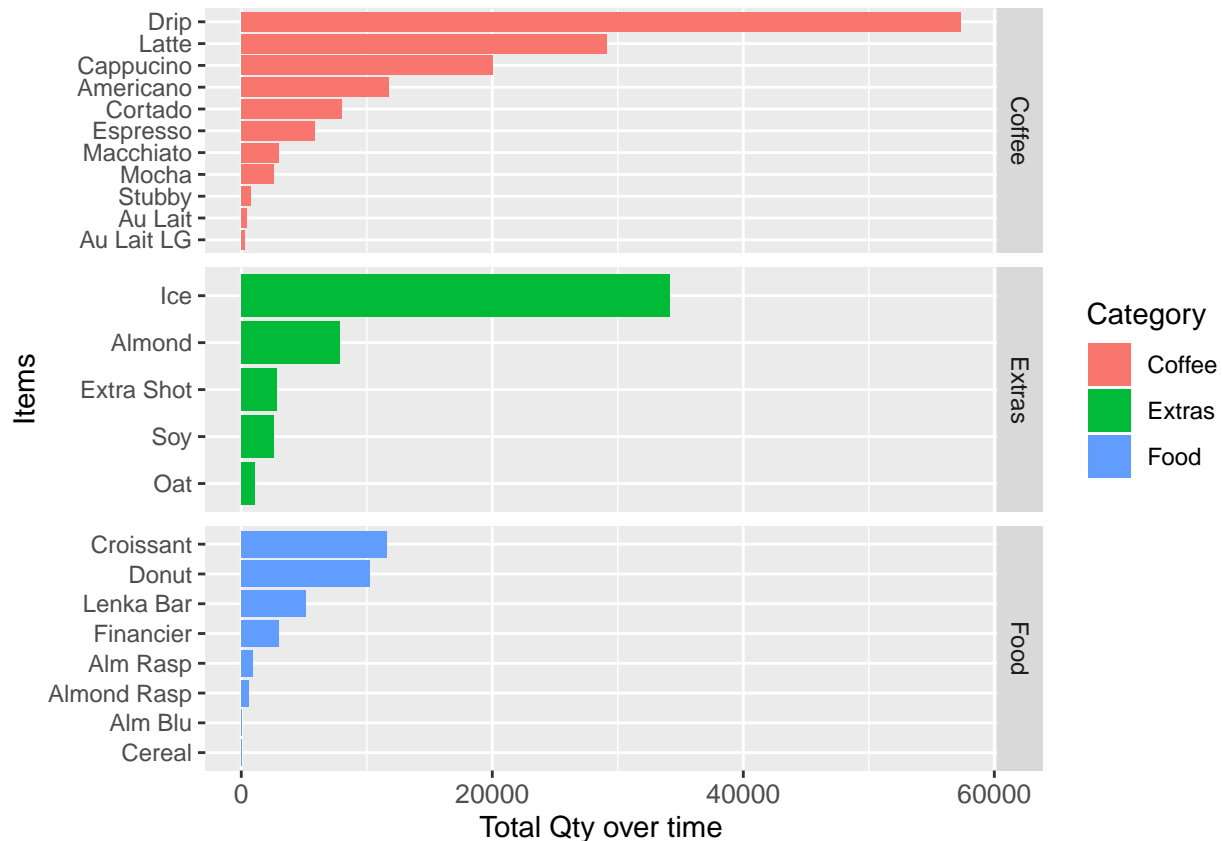
### Interpretation

At a glimpse, Central Perk's business focuses on Coffee and Food categories, which doesn't come as a surprise! Oddly, we also see high sales in Extras during the month of July and August. So let's narrow down to only Coffee and Food along with Extras and see what is going on.

### Sales Volume Analysis by Items

Now, we look at most sold items in each of three categories i.e. Coffee, Food and Extras

```
data %>%
  filter(Category %in% c('Coffee', 'Extras', 'Food')) %>%
  group_by(Category, Item) %>%
  summarise(Tol_qty = sum(Qty)) %>%
  ggplot(aes(x = reorder(Item, Tol_qty), y = Tol_qty, fill = Category)) +
  geom_bar(stat = 'identity') +
  theme(axis.text.x = element_text(size = 10)) +
  coord_flip() +
  facet_grid(Category ~ ., scales = "free_y") +
  labs(x = "Items", y = "Total Qty over time")
```



### Interpretation

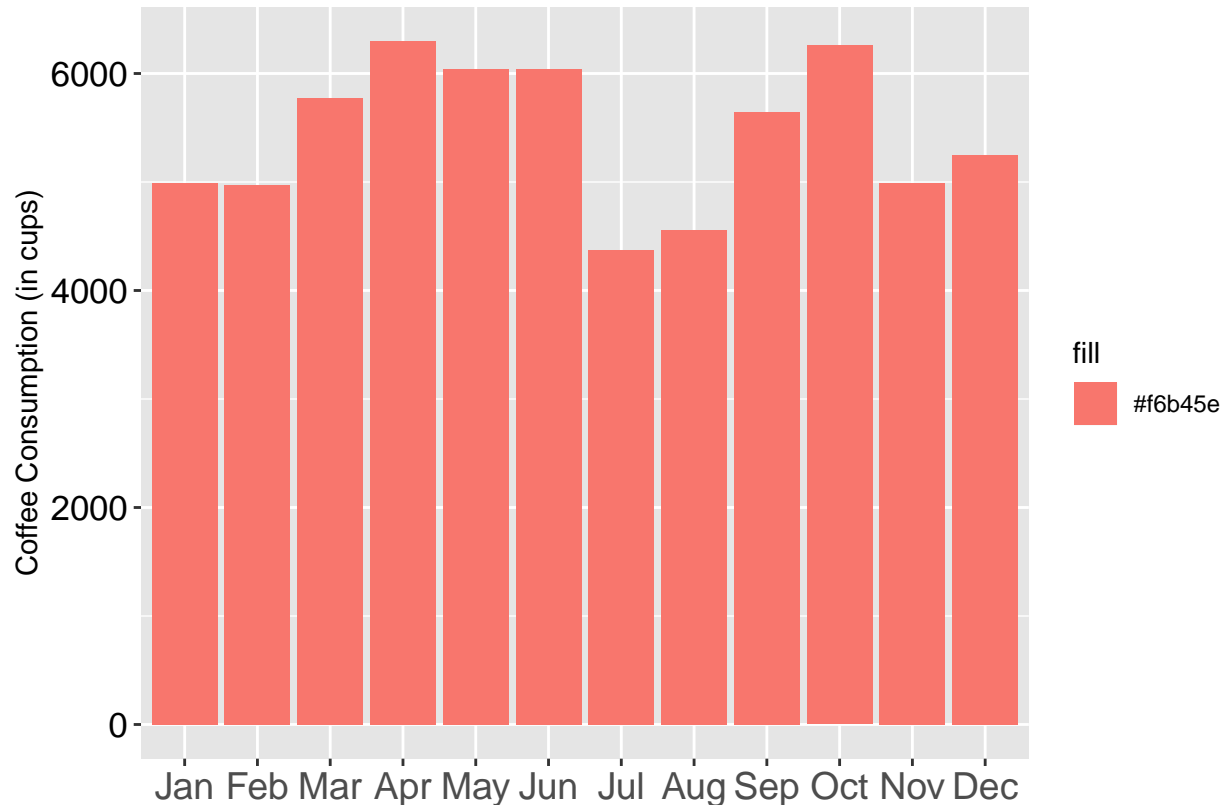
Apparently, Drip is the most popular product in the coffee shop, Latte comes second. And for food, Croissant and Donut stand out among others. In Extras, Ice shows a significantly high sales as compared to other items. We will now further look into the patterns in Coffee and Extras by Month.

### Sales Volume Analysis of Coffee By Month

Monthly sales are analyzed to see the seasonality and trends in sales.

```
data %>%
  filter(Category == 'Coffee') %>%
  mutate(month = lubridate::month(Date, label = TRUE)) %>%
  mutate(year = lubridate::year(Date)) %>%
  group_by(month, year) %>%
  summarise(Qty = sum(Qty)) %>%
  ggplot(aes(month, Qty, fill = '#f6b45e')) +
```

```
stat_summary(fun.y = mean, geom = "bar", na.rm = TRUE) +
labs(x = "", y = "Coffee Consumption (in cups)") +
theme(panel.background = element_rect(fill = "grey90"),
      axis.text.y = element_text(size = 13, colour = "black"),
      axis.text = element_text(size = 14))
```



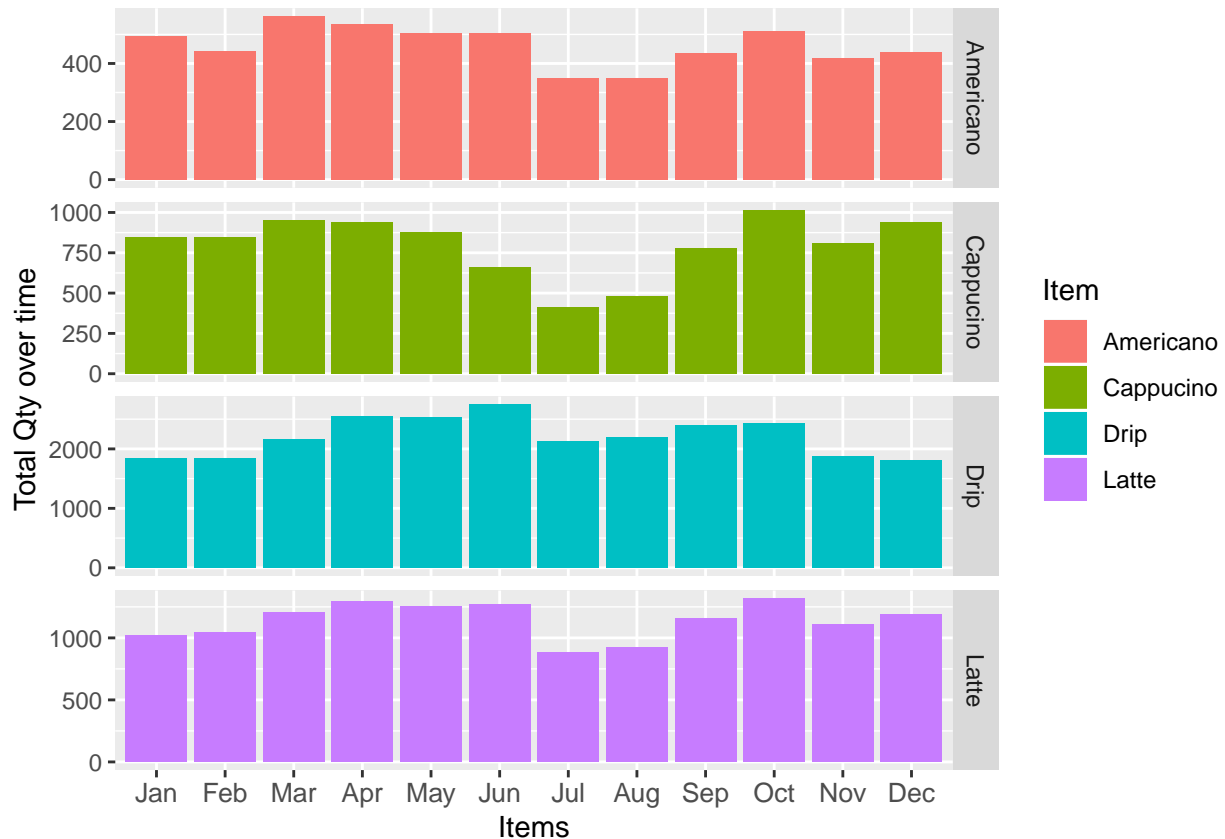
### Interpretation

Across a year, Central Perk has its best business for Coffee in the Spring and Fall, but business cools down in the months of July and August. This seasonality pattern doesn't seem reasonable for a coffee shop. Let's look at which coffee type is causing this decline. Is it a hot/cold coffee?

*Sales Volume Analysis of each major Coffee (Drip, Latte, Cappuccino, Americano) type By Month* Let's analyse if there is any specific type of coffee causing this drop during the months of July and August or is the drop is general?

```
data %>%
  filter(Item %in% c('Drip', 'Latte', 'Cappuccino', 'Americano')) %>%
  mutate(month = lubridate::month(Date, label = TRUE)) %>%
  mutate(year = lubridate::year(Date)) %>%
  group_by(month, year, Item) %>%
  summarise(Tot_qty = sum(Qty)) %>%
  ggplot(aes(x = month, y = Tot_qty, fill = Item)) +
  stat_summary(fun.y = mean, geom = "bar", na.rm = TRUE) +
  theme(axis.text.x = element_text(size = 10)) +
  facet_grid(Item ~ ., scales = "free_y") +
  labs(x = "Items", y = "Total Qty over time")
```





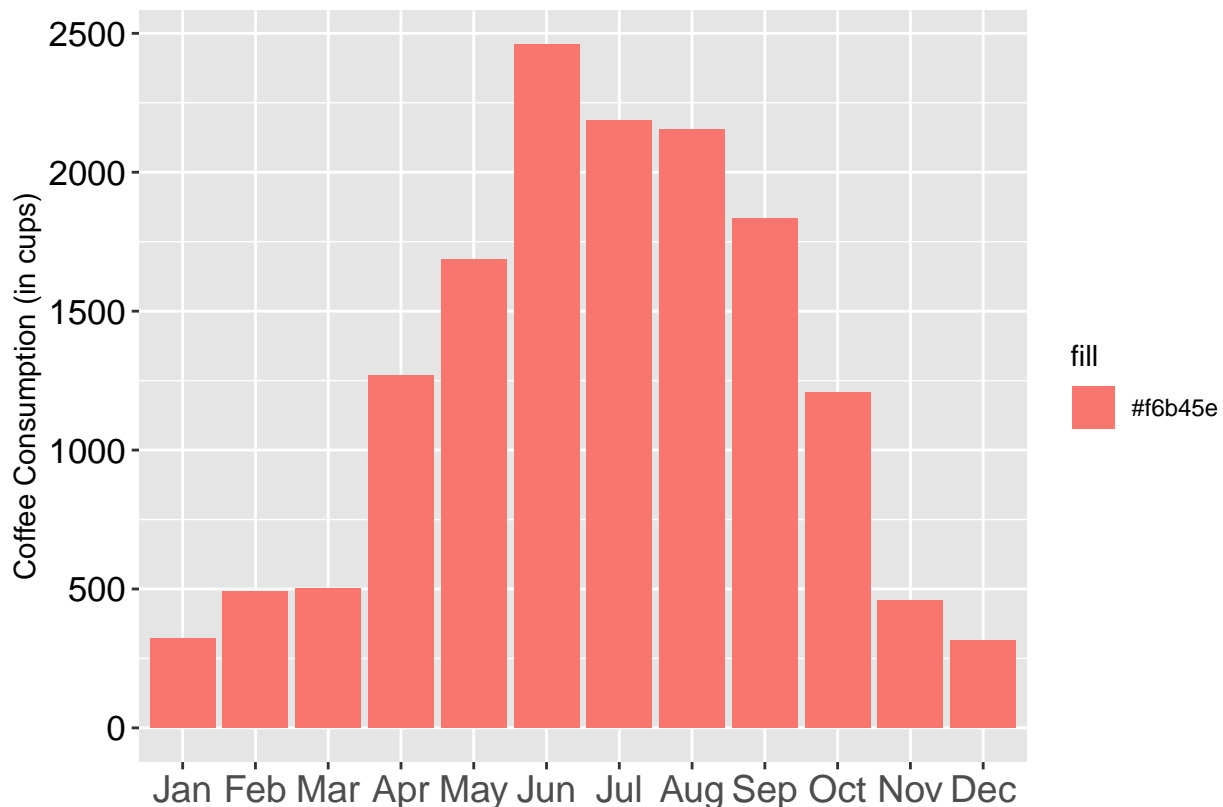
### Interpretation

From the above graph we can see that sales of every coffee drops during the month of July and August. These are usually the summer months. We have an intuition that can be related to heat and Ice. Let's look at the distribution of 'Ice' by months.

### Sales Volume Analysis of Ice By Month

Monthly sales are analyzed to see the seasonality and trends in sales.

```
data %>%
  filter(Item == 'Ice') %>%
  mutate(month = lubridate::month(Date, label = TRUE)) %>%
  mutate(year = lubridate::year(Date)) %>%
  group_by(month, year) %>%
  summarise(Qty = sum(Qty)) %>%
  ggplot(aes(month, Qty, fill = '#f6b45e')) +
    stat_summary(fun.y = mean, geom = "bar", na.rm = TRUE) +
    labs(x = "", y = "Coffee Consumption (in cups)") +
    theme(panel.background = element_rect(fill = "grey90"),
          axis.text.y = element_text(size = 13, colour = "black"),
          axis.text = element_text(size = 14))
```



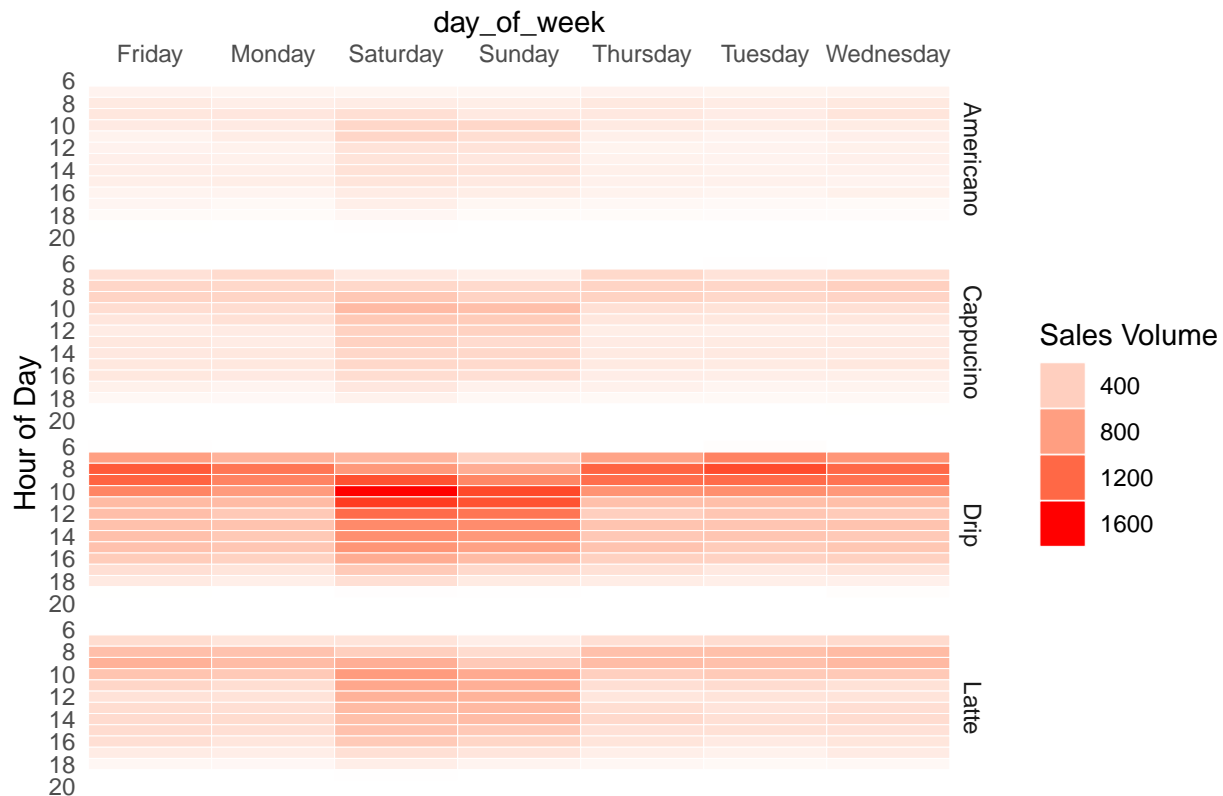
#### Interpretation

This shows us that due to the increase in temperature, people prefer adding Ice to their coffee. But as we also see a decrease in sales of coffee during these month, our intuition says that coffee sales might have dropped because adding Ice to the coffee costs money. This is bad for client. As by gaining some profits from Ice, we are losing on a lot more profit by selling less coffee during the warmer months.

#### Analysis of coffee sales by time of day and day of week

```
data %>%
  mutate(hour = lubridate::hour(Time)) %>%
  filter(Item %in% c('Americano', 'Latte', 'Cappuccino', 'Drip')) %>%
  group_by(day_of_week, hour, Item) %>%
  summarise(Qty = sum(Qty)) %>%
  ggplot(mapping = aes(x = day_of_week, y = hour, fill = Qty)) +
  geom_tile(colour = "white") +
  guides(fill = guide_legend(title = "Sales Volume")) +
  scale_fill_continuous(low = 'white', high = 'red') +
  scale_y_reverse(breaks = seq(20,6,-2), labels = seq(20,6,-2), expand = c(0,0)) +
  scale_x_discrete(expand = c(0,0), position = "top") +
  labs(title = "Average Coffee Sales Volume by Day of Week / Hour of Day",
       y = "Hour of Day") +
  facet_grid(Item ~ .) +
  theme_bw() + theme_minimal() +
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())
```

## Average Coffee Sales Volume by Day of Week / Hour of Day



### Interpretation

Drip Small at Central Perk sells like hotcakes, especially in the morning from 7am to 10am during Weekdays and 9am to 15pm on Weekends. We might want to prepare ahead of time in terms of material and staffing to smooth the demand.

*Analysis on low profit products* We looked at products which might be causing a loss to the business and we found that beer is being sold at a very low frequency and generating a loss for the business.

```
data %>%
  filter(Category == 'Beers') %>%
  group_by(Item) %>% summarise(Total_Profit = sum(Profit)) %>%
  arrange(Total_Profit)
```

```
## # A tibble: 10 x 2
##   Item                Total_Profit
##   <chr>                <dbl>
## 1 Lagunitas IPA        -152.
## 2 Six Point Pilsner    -46.6
## 3 Blue Point Lager    -16.8
## 4 Crisp                -14.7
## 5 Purple Haze         -10.7
## 6 Sierra Nevada PA    -5.58
## 7 Anchor               4.41
## 8 Abita Amber          4.9
## 9 Lagunitas            5.86
## 10 Ommegang             7.35
```

## Recommendations

Below are some recommendations from the insights that our team got from the data. These recommendations are made keeping in mind that the client is able to implement them across all the coffee stores. Our recommendations are easy to implement and focus directly on increasing profits.

*To increase 'loyal' customers rather than just registered customers, business should take following actions:*

1. Incentivizing repeat visits: Introducing an offer on their card, where each coffee purchase earns customers 10 points and they can redeem a free coffee when they collect 100 points. This can be implemented until they become frequent visitors to the store.
2. Sending promotion emails: Sending emails to customers who have not made a purchase with us in last 3 months would provide another incentive for customers to keep visiting the shop.

*To increase sales by making sure that business is providing the right items to customers at right point of time, business should take following actions:*

1. Leverage tea and non-cafeinated drinks in winter: As customers are much more likely to buy tea and non-cafeinated drinks in winter evenings, client should introduce new and special tea, and non-cafeinated drinks. Since customers are more likely to purchase products they see, these items should be displayed on the front of the menu. Posters of novelty items can be displayed outside the store to attract customers. As our customers purchase a lot of beverages and they are likely to try the new drinks, pricing these new introductions slightly higher will drive more profits.
2. Increase sales of coffee during afternoon: Most of the coffee and food are being sold in the morning and sales drop a lot in afternoon and evening, especially in the summer. This makes sense as people tend to consume less coffee last in the day. To attract customers during afternoon and maintain constant demand, client should introduce happy hours from 4 pm to 7 pm, except the weekends. A 10% discount on coffee will attract customers, increase sales and still make a profit on another 10%.
3. Increase sales of food: Registered customers like our food and non-registered customers are less likely to buy our food. This indicates that our food has good taste but not attractive to new customers. Client should keep the taste of food same but make the appearance more attractive. Also, client should put food in the center of menu and display cabinet. Staff should recommend food to new customers.
4. Give free Ice during summer months: Sales of every coffee drops in July and August i.e. summer months. This is because our client is charging extra for the ice. As customers like to drink cold coffee in summer, client should stop charging for ice during these months. Profits made by selling more coffee will easily cover the cost of providing free ice.
5. Stop selling beer: Looking at the past sales, all the products in beer category are causing a loss to the business. Client would have introduced beers in the past to attract customers, but selling it at a low price is not helping the sales of other products. Our recommendation of 10% off on coffee during afternoon will help achieve this objective. Hence, client should stop selling beers. This will also help in reducing inventory cost or can help support inventory for other items.