QVC Fulfillment –

Does Speed Matter in E-Commerce

Group Members:

Evan Canfield

Khabirul Gainey

Joshua Ganz

Jake Hoertt

Julian Mucha

Harika Mudigonda

August 7, 2019

Summary

QVC is a televised shopping service, broadcasting live retail programming 24 hours a day. It's a virtual shopping center, offering a vast selection of products including electronics, apparel, health and beauty, jewelry, and home decor.

In the US alone, QVC has a reach of 96 million households and ships millions of packages each year. As with any large company offering products, fulfilling customer orders in a timely manner is essential, not only for QVC's reputation, but to their survival as a company. Quick turnaround on shipping goods to customers involves an extensive logistics and delivery network, which, in turn, can lead to greater costs. Besides offering quality products, it is imperative for QVC to focus on the importance of overall experience, from the time a customer places an order until receipt of purchase.

Having analyzed QVC's customer geography, distribution network, product mix, and purchase patterns, we explored the following questions:

1. Does the current distribution network maximize customer penetration (spend)? If not, what should QVC do to increase customer penetration with the current distribution network?
2. Are their specific product categories that should be located in specific distribution centers?
3. Do customers that receive their product sooner purchase more than customers with longer delivery times? In other words, do customers become repeat purchasers if there is a short fulfillment time?

Analysis

Question 1

Does the current distribution network maximize customer penetration (spend)? If not, what should QVC do to increase customer penetration with the current distribution network?

In order to evaluate whether the current QVC distribution network is maximizing customer penetration, it is necessary to understand what the current customer penetration is. Figure 1 shows QVC sales by state, with the percentages shown being the percent each state accounts for against the total. Sales are also made to Alaska, Hawaii, and Puerto Rico, but these states and territories account for a small percentage of overall sales (less than 0.5% combined), and are therefore excluded from the image to improve legibility.

Figure 1: Sales By State

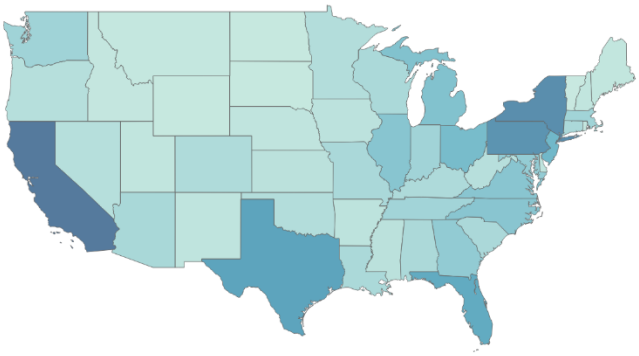


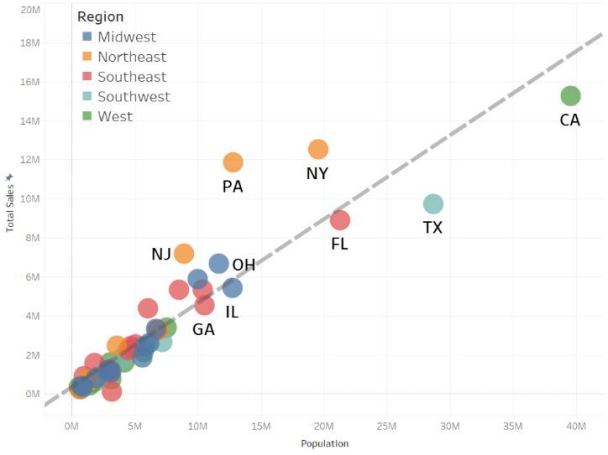
Table 1 shows top five states by percentage of total sales. The states in the table are not surprising, as they match up with the top five states by population, per the US Census (Ref. 2.1). What is unexpected is the relative ranking of each state. Pennsylvania has a greater percentage of sales than Texas, even with Texas having more than twice as many people.

Table 1: Top 5 States by Sales

State	Percent of Total Sales	Population
California	9.56%	39,557,045
New York	7.82%	19,542,209
Pennsylvania	7.43%	12,807,060
Texas	6.04%	28,701,845
Florida	5.54%	21,299,325

Figure 2 more clearly shows the relationship between state sales and population.

Figure 2: Sales vs. Population, By State

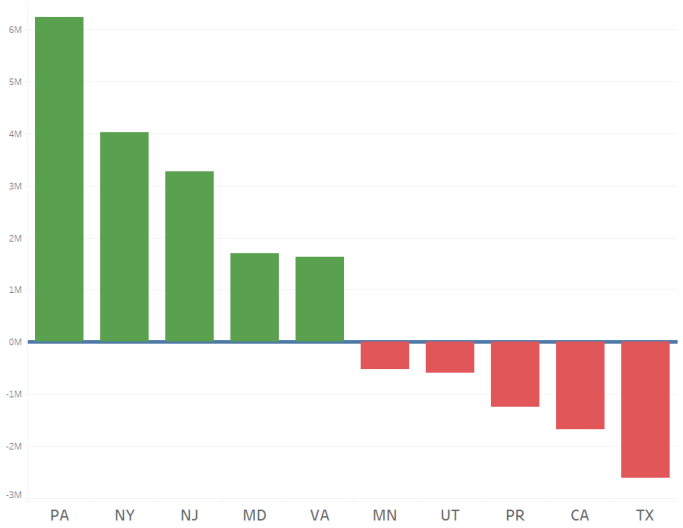


The states listed in Table 1 are clearly visible in the figure above. The relationship between population and sales appears linear. A regression analysis between sales and population confirms this relationship, and that it is statistically significant.

The linear relationship is represented by the trend line on Figure 2, with the slope of the trend line (0.42) approximating the expected sales in a state based on the state's population. State's that are above this trend line (i.e.: NJ, PA, NY) are exceeding this baseline performance. State's below this line (i.e.: CA, TX, FL) are underperforming.

Figure 3 further visualizes which states are exceeding this baseline performance and which are falling short, expressed in total dollars. Florida, while not within the bottom 5, is the seventh lowest performer.

Figure 3: Sales Differential - Top and Bottom 5



To understand why states like California and Texas are underperforming, while states like New York and Pennsylvania may be over performing, it is instructive to look at the relationship between a state's per capita sales and the average fulfillment time, that is, the time from a customer ordering an item to receiving it. Figure 4 and Figure 5 explore this relationship.

While the linear relationship is not as strong as in Figure 2, the relationship between sales per capita and fulfillment time shown in Figure 4 still shows a trend of decreasing sales per capital as average fulfillment time increases.

Figure 4: Sales per Capita vs. Fulfillment Time

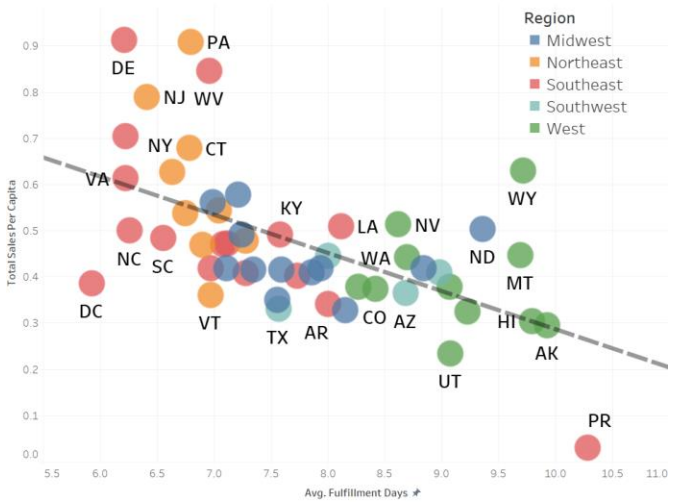
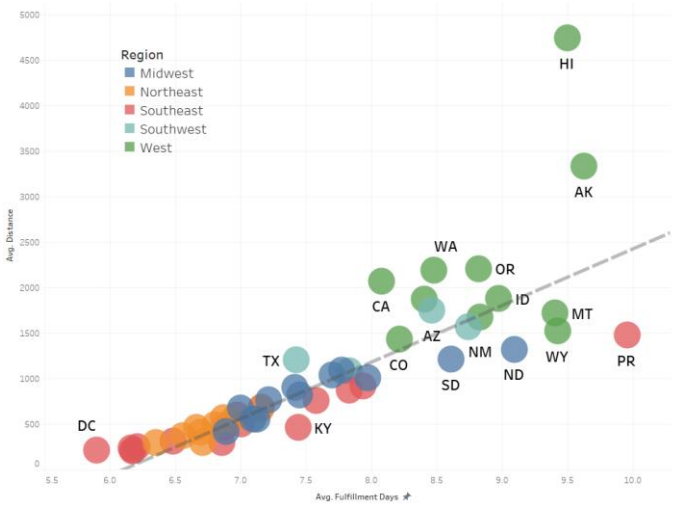


Figure 5 shows the relationship between fulfillment time and the distance between a shipped items origin and destination. Again, a linear trend,

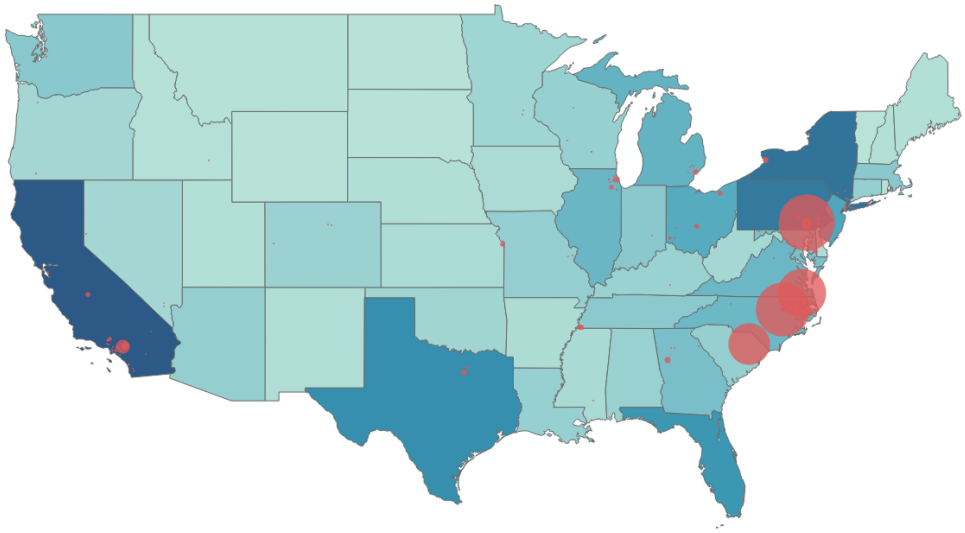
Figure 5: Distance vs. Fulfillment Time



Both Figure 4 and Figure 5 color states by geographical region (Midwest, Northeast, Southeast, Southwest, and West). In both plots you see a general pattern in the clustering of regions, the more eastern regions are more to the left of the plot (shorter fulfillment and distance) and the more western regions are more to the right of the plot (longer fulfillment and distance).

This clustering behavior in the regions shown in Figure 4 and Figure 5 is a result of the QVC distribution network, as shown in Figure 6. Distribution centers are shown as circles, with the total amount of sales that pass through each one relative to the size of the circle. As Figure 6 shows the majority of QVC purchases (91.2%) are shipped out of one of four distribution centers, all on the East Coast. The largest distribution center in California, #0125, processed 1.7% of all sales. The largest distribution center in Texas, 1914, processed 0.31% of all sales.

Figure 6: Distribution Center Locations

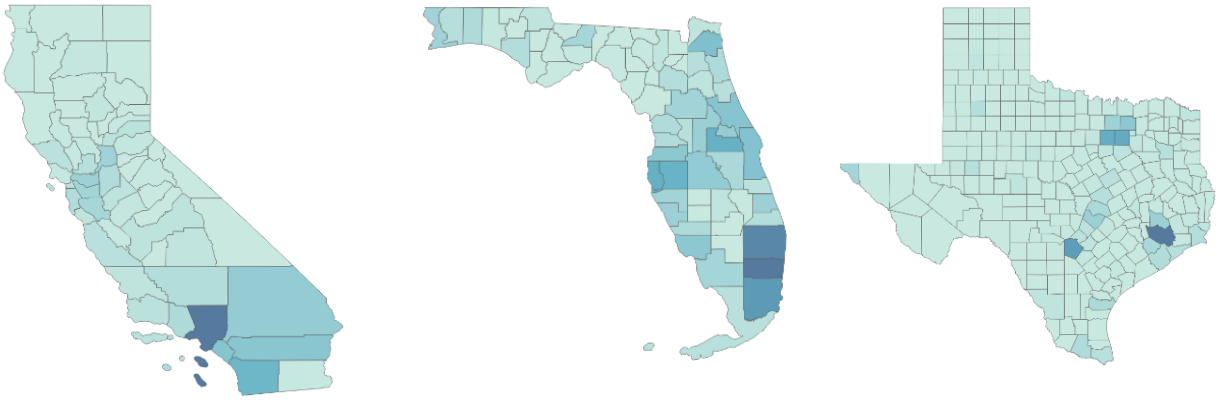


To increase penetration in possible high return states, that is, high population states, shipping times to these states must be decreased. This would require improving the distribution network so that large hubs exist in California and Texas. Florida, while not as low performing as California or Texas, should also be considered. The fulfillment times in Florida are less than Texas and California, but with such a large population the potential for return is greater than some smaller states with low performance, such as Utah or Minnesota. In addition, Florida is the closest state to Puerto Rico, which ranks last in average fulfillment time and greatly underperforms compared to baseline sales. As Puerto Rico is not part of the continental United States, and cannot rely on the same shipping methods, further study would be needed the extent an additional distribution center would affect shipping times.

The following figure contains county level sales within California, Florida and Texas. The maps are to help recommend where in each state a new distribution center should be placed.

Figure 7: County Level Sales - California, Florida, and Texas

California **Florida** **Texas**



Recommendations

- California: Los Angeles County
    - Contains Los Angeles metro area
- Florida: Broward County
    - Contains Fort Lauderdale, just north of Miami
- Texas: Harris Country
    - Contains Houston metro area

Question 2

Are their specific product categories that should be located in specific distribution centers?

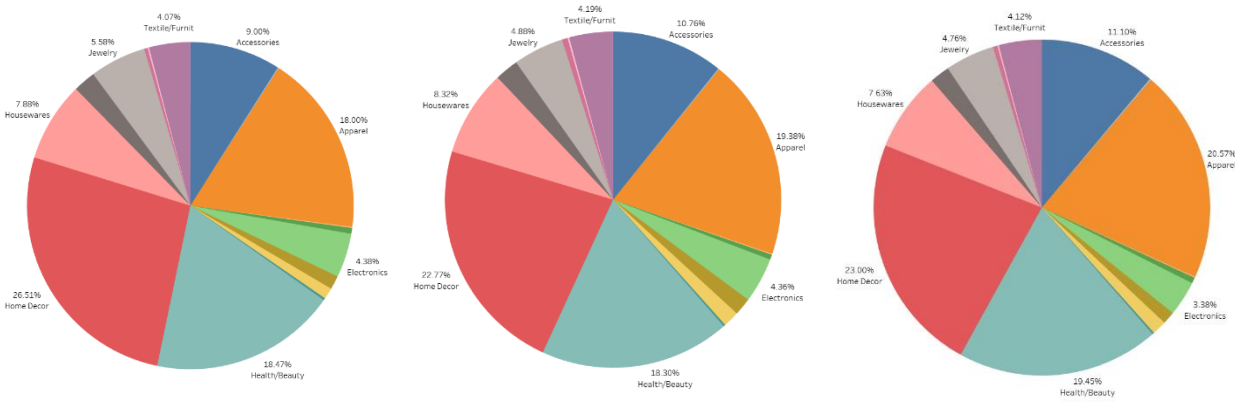
To determine whether specific product categories should be in specific distribution centers, it was important to identify the states that have a rate of low sales compared to its population. Based on Figure 2 and the analysis from question 1, Florida, California, and Texas had underperforming sales per each state's population. In addition, the difference between expected and actual sales per each state's population revealed states exceeding and not exceeding baseline sales (Figure 3). Texas and California underperformed the most, while Florida was the seventh most underperforming out of all states. Lastly, the sales per capita

compared with fulfillment time and distance compared with fulfillment time gave insights into the relationships. Generally, as sales per capita decreases, fulfillment time increases; as distance increases, so does fulfillment time (Figures 4 and 5). Taking into consideration shipments from distribution center locations (Figure 6), most shipments came from one of four centers on the east coast. Since a goal for QVC is to increase penetration in more populous states, having a shorter distance and/or quicker fulfillment time could impact an increase in sales. Therefore, acknowledging Florida, Texas, and California are underperforming, investigating the percentage of product categories, shipped to these three states, assists in determining what product categories should be stocked in local distribution centers to reduce fulfillment time.

In Figure 8, the top product categories for Florida, Texas, and California are: Apparel, Health/Beauty, Home Décor, and Housewares. These categories represent roughly 70% of all products bought by customers in those states. If the above-mentioned categories were most of the stocking in the local distribution centers, it would help to reduce the fulfillment time. However, knowing what individual products were mostly purchased within those selected categories would also be valuable information. With more accurate, granular data, further study to analyze percentage of different products within categories could give more specific insight as to what products should be stocked locally.

Figure 8: Product Categories Based on Products Shipped

Florida **Texas** **California**



Exploring some more, Association Rule Mining was used to see if there were any patterns, or rules, between products. There were 611 unique rules discovered and five frequent sets of two. However, the support used to generate any rules was extremely low at 0.000005. Conclusively, it was determined that stocking local distribution centers with the associated products would not be advantageous.

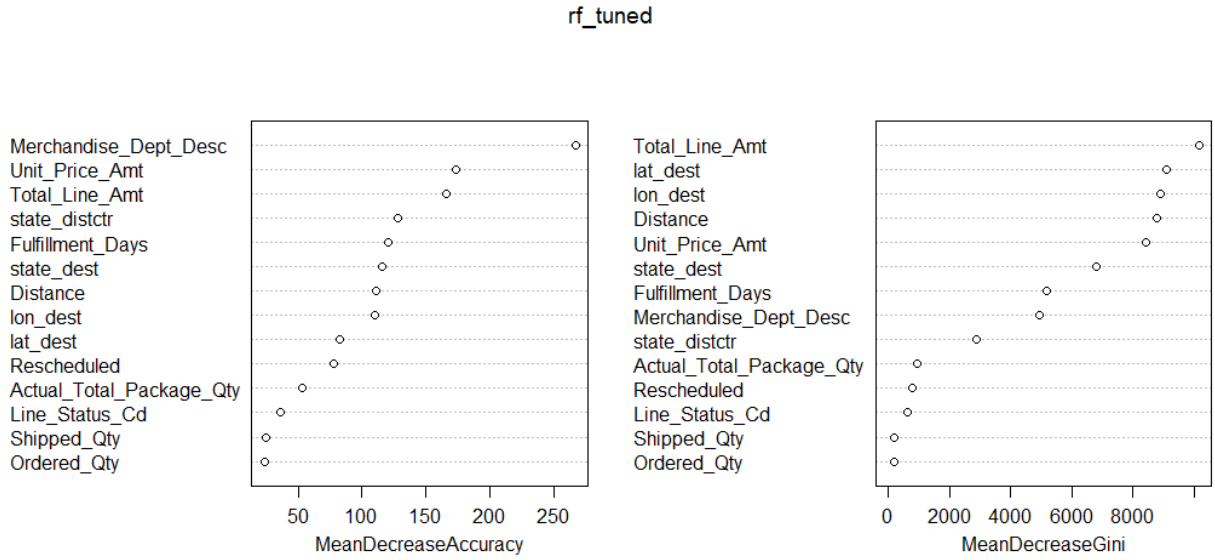
Question 3

Do customers that receive their product sooner purchase more than customers with longer delivery times? In other words, do customers become repeat purchasers if there is a short fulfillment time?

The first model we utilized was a logistic regression model. We decided to approach this problem as a classification problem. After removing variables that showed collinearity such as **Distance** and **Fulfillment_Days** and various other insignificant variables, the logistic regression model consisted of six variables. Five of these variables were numerical in nature. These variables were: **Total_Line_Amt**, **Unit_Price_Amt**, **Actual_Total_Package_Qty**, **Rescheduled** which was binary in nature, and **Fulfillment_Days**. The logistic regression utilized one categorical variable: **Merchandise_Dept_Desc**. This variable consisted of 20 categories which were all included in the logistic regression. The importance of each variable in the logistic regression can be found below.

```
Optimization terminated successfully.
         Current function value: 0.663685
         Iterations 5
                                Results: Logit
================================================================================
Model:                     Logit                 Pseudo R-squared:       0.042
Dependent Variable:        y                     AIC:                    308665.3297
Date:                      2019-08-04 10:16      BIC:                    308913.8895
No. Observations:          232503                Log-Likelihood:         -1.5431e+05
Df Model:                  23                    LL-Null:                -1.6113e+05
Df Residuals:              232479                LLR p-value:            0.0000
Converged:                 1.0000                Scale:                  1.0000
No. Iterations:            5.0000
--------------------------------------------------------------------------------
                                    Coef.  Std.Err.    z     P>|z|   [0.025  0.975]
--------------------------------------------------------------------------------
Total_Line_Amt                     -0.0029   0.0003  -8.7745 0.0000 -0.0035 -0.0022
Unit_Price_Amt                      0.0011   0.0004   3.0509 0.0023  0.0004  0.0018
Actual_Total_Package_Qty            0.1613   0.0086  18.8232 0.0000  0.1445  0.1781
Rescheduled                        -0.1630   0.0094 -17.3426 0.0000 -0.1814 -0.1446
Fulfillment_Days                    0.0079   0.0004  18.4277 0.0000  0.0071  0.0088
Merchandise_Dept_Desc_Accessories   0.1329   0.0185   7.1786 0.0000  0.0966  0.1692
Merchandise_Dept_Desc_App/Accss Event 0.6885 0.1497  4.6002 0.0000  0.3951  0.9818
Merchandise_Dept_Desc_Apparel       0.6643   0.0187  35.5452 0.0000  0.6277  0.7009
Merchandise_Dept_Desc_Collectibles  0.4841   0.1459   3.3180 0.0009  0.1981  0.7701
Merchandise_Dept_Desc_Costume Jewelry 0.0981 0.0730  1.3448 0.1787 -0.0449  0.2412
Merchandise_Dept_Desc_Electronics  -1.0161   0.0235 -43.1802 0.0000 -1.0622 -0.9699
Merchandise_Dept_Desc_Entertainment -0.1004 0.0334  -3.0043 0.0027 -0.1659 -0.0349
Merchandise_Dept_Desc_Fun & Leisure -0.2229 0.0347  -6.4310 0.0000 -0.2909 -0.1550
Merchandise_Dept_Desc_Health        0.0481   0.0869   0.5537 0.5798 -0.1222  0.2184
Merchandise_Dept_Desc_Health/Beauty -0.0502  0.0148  -3.3974 0.0007 -0.0792 -0.0213
Merchandise_Dept_Desc_Home Decor    0.1622   0.0151  10.7755 0.0000  0.1327  0.1917
Merchandise_Dept_Desc_Housewares   -0.3313   0.0186 -17.7682 0.0000 -0.3679 -0.2948
Merchandise_Dept_Desc_IQVC Divisional -0.5355 0.0298 -17.9943 0.0000 -0.5938 -0.4771
Merchandise_Dept_Desc_Jewelry       0.2121   0.0252   8.4091 0.0000  0.1627  0.2616
Merchandise_Dept_Desc_License Hardgds -0.7103 0.0630 -11.2711 0.0000 -0.8338 -0.5868
Merchandise_Dept_Desc_PUBLIC RELATION -0.2030 0.1433  -1.4165 0.1566 -0.4839  0.0779
Merchandise_Dept_Desc_Returns      -0.5391   0.6463  -0.8342 0.4042 -1.8058  0.7275
Merchandise_Dept_Desc_Textile/Furnit -0.1246 0.0226  -5.5075 0.0000 -0.1690 -0.0803
Merchandise_Dept_Desc_UNKNOWN       0.7274   0.4543   1.6011 0.1094 -0.1630  1.6178
================================================================================
```

The second model our group utilized was a random forest model to derive insight from the data set. Our primary goal with the random forest model was to identify variables within the data set that most influence a first-time order becoming a repeat customer. In order to do this, we wanted to observe the variable importance plot from the tuned Random Forest Model. This model had 1000 trees and used three variables from the sample set for each tree. The variable importance plots can be seen below.



Question 3 probes us to explore if customers who receive their packages first will purchase more in the future. Based on the graph above, we observe fulfillment days to be a top 7 contributing variable for both accuracy and Gini. Therefore, we observe that fulfillment days is an important factor is predicting if a customer will be a repeat customer, but other variables such as unit price and total price paid rank higher in both metrics. We can see distance is also an important variable, but it has strong positive correlation with fulfillment days, so no additional conclusions are made from it.

Data Processing

The processing of data for this project was primarily done with R (Ref. 1.1). All subsequent code provided is in R.

Data Inputs - Provided

The following files were the provided Option 2 of the DSBA 6211 Group Project:

File	Description
QVC Data 1.xlsx	QVC purchase records
QVC Data 2.xlsx	QVC purchase records
QVC Data 3.xlsx	QVC purchase records
QVCdist_ctr	Geographical information on the QVC distribution centers. Data set name: **qvc_distctr**
QVCorderstatustype.xlsx	Order type information
QVC Data Dictionary.xlsx	Data dictionary describing the variables in QVC Data 1, 2, and 3.

The files *QVC Data 1.xlsx*, *QVC Data 2.xlsx*, and *QVC Data 3.xlsx* constitute a single large data set broken into three separate files.

The QVCorderstatustype.xlsx file was not used in this analysis. The QVC Data Dictionary functioned simply as a reference document and as not directly used in any of the subsequent analysis.

Data Inputs - Supplemental

State Population Estimates

State population data was provided by 2018 National and State Population Estimates from the US Census Bureau (Ref. 2.1). The raw data from the Census Bureau was then processed so that only the state code, state (and state equivalents), and 2018 population estimate were in the file used in this analysis.

The name for the data set within the provided R code is **us_census_2018**.

Zip Codes

The **zip_codes** data set from the noncensus package (Ref. 3.6) was used to provide city, state, latitude, and longitude data for both the origin (QVC distribution center) and destination points. Locations for both origin and destination were approximated as to the zip code of each point. Additionally, the **zip_codes** data set provided the Federal Information Processing Standard (FIPS) code for each destination zip code. FIPS codes are unique identifiers for US counties (and county equivalents).

The name for the data set with in the provided R code is **zip_codes**.

FIPS Codes

A FIPS code is a unique code identifying a US state and county. FIPS codes used in this analysis are provided by **fips_codes** data set from the tidycensus package (Ref. 3.13).

The name for the data set within the provided R code is **fips_codes**.

Processing the Data

Data processing was primarily done through R. The following packages (Section 1.3) were used: **arules, arulesViz, caTools, dplyr, tidyr, geosphere, lubridate, noncensus, readxl, ROCR, ROSE, tidycensus,** and **stringr.**

Before upload and processing in R an error was noticed. The rows listed below had blank value inserted at column **Size_DESC**. The remaining data in that row was then shifted over to the right by one column. This data was readjusted manually.

- QVC Data 1.xlsx: 220464 (#Sales_Order_Nbr: 649344509406)
- QVC Data 1.xlsx: 259651 (#Sales_Order_Nbr: 449320937874)
- QVC Data 2.xlsx: 111684 (#Sales_Order_Nbr: 649288839226)
- QVC Data 2.xlsx: 462659 (#Sales_Order_Nbr: 849460177698)
- QVC Data 3.xlsx: 271093 (#Sales_Order_Nbr: 149349466481)
- QVC Data 3.xlsx: 303069 (#Sales_Order_Nbr: 649356909726)

Once uploaded into R the QVC sales data was combined into a single data frame. The name for the combined data set within the provided R code is **qvc_data**.

Initial Processing

The variable **#Sales_Order_Number** variable was renamed to **Sales_Order_Number** to remove the octothrop. Furthermore, **SHIP_TO-ZIP** was renamed **zip_dest** to be consistent with data that will be added to the data frame during a later step.

```
qvc_data <- qvc_data %>%
  rename(Sales_Order_Nbr = '#Sales_Order_Nbr',
         zip_dest = SHIP_TO_ZIP)
```

After studying the project questions and the provided data, the following variables were determined to not be relevant to the analysis were then dropped.

Note: The information from **SHIP_TO_STATE** is important to the analyzing the data, but will be repopulated through the **zip_codes** data set during a later step in the process.

```
drop_variables = c(
  "Sales_Order_Line_Nbr",
  "Order_Type_Cd",
  "Shipping_Priority_Ind",
  "Line_Status_Dt",
  "Skn_Id", "Sku_Id",
  "Color_Desc","Size_Desc",
  "Assigned_Dc_Id",
  "Cancelled_Qty",
  "Merchandise_Div_Desc",
  "Carrier_Used_Tracking_Id",
  "Shipment_Status_Dt",
  "Pickup_Dt",
  "Scheduled_Delivery_Dt",
  "Package_Scan_Dttm",
  "Package_Cnt",
  "SHIP_TO_CITY",
  "SHIP_TO_STATE"
  )
qvc_data <- qvc_data %>%
  select(-drop_variables)
```

Modifying the Data

As the provided data was in .XLSX form not all variables had the correct data types following upload. The following modifications were necessary.

Changing Data Types

The variable **Merchandise_Dept_Desc** was imported as a character, but functions as a factor. **Sales_Order_Nbr**, **Party_Id** , **Product_Id** , and **Package_Id** are all identification codes, and although are comprised of numbers, function as a character strings.

```
qvc_data <- qvc_data %>%
  mutate(
    Merchandise_Dept_Desc = factor(Merchandise_Dept_Desc),
    Sales_Order_Nbr = as.character(Sales_Order_Nbr),
    Party_Id  = as.character(Party_Id),
    Product_Id = as.character(Product_Id),
    Package_Id = as.character(Package_Id)
    )
```

Leading Zeros

Several numeric identification codes were uploaded as numeric values and not character strings. This resulted in leading zeros for these variables being dropped.

To properly use these variables the leading zeros need to be reinserted, and the variables converted to character type, in both the QVC sales data and QVC Distribution Center data sets. Additionally, leading zeros for FIPS code in the **zip_codes** data set were also dropped, and therefore needed to be reinserted.

```
# Source_Ship_Warehouse_Nbr - QVC Data
qvc_data$Source_Ship_Warehouse_Nbr  <- str_pad(string =
qvc_data$Source_Ship_Warehouse_Nbr,
                                  width = 4,
                                  side = "left",
                                  pad = "0")

# Source_Ship_Warehouse_Nbr - QVC Distribution Center
qvc_distctr$Source_Ship_Warehouse_Nbr  <- str_pad(string =
qvc_distctr$Source_Ship_Warehouse_Nbr,
                                  width = 4,
                                  side = "left",
                                  pad = "0")

# zip_dest
qvc_data$zip_dest  <- str_pad(string = qvc_data$zip_dest,
                                  width = 5,
                                  side = "left",
                                  pad = "0")

# FIPS
zip_codes$fips  <- str_pad(string = zip_codes$fips,
                                  width = 5,
                                  side = "left",
                                  pad = "0")
```

Connecting Data Sets

The QVC sales data set was then combined with two other data sets, the QVC Distribution Center data and the **zip_codes** data set. The POSTL_CD variable from the QVC Distribution Center data set was also renamed to maintain a consistent naming structure.

The **zip_codes** data set was joined to provide city, state, latitude, and longitude data for the QVC warehouse and the package destination based on the provided zip codes, as well as FIPS codes based on the destination zip codes. Variables were renamed to provide a clean and consistent naming convention.

```
qvc_data <- qvc_data %>%
  left_join(
    select(qvc_distctr,Source_Ship_Warehouse_Nbr,POSTL_CD),
            by = "Source_Ship_Warehouse_Nbr"
            ) %>%
  rename(zip_distctr = POSTL_CD)

qvc_data <- qvc_data %>%

  #Join zip_codes to Distribution Center zip code
  left_join(
    select(zip_codes, zip:longitude),
    by = c("zip_distctr" = "zip")
  ) %>%

  #Rename Distribution Center Variables
  rename(
    city_distctr = city,
    state_distctr = state,
    lat_distctr = latitude,
    lon_distctr = longitude
  ) %>%

  #Join zip_codes to Destination zip code
  left_join(zip_codes
            , by = c("zip_dest" = "zip")) %>%

  #Rename Destination Columns
  rename(
    city_dest = city,
    state_dest = state,
    lat_dest = latitude,
    lon_dest = longitude,
    fips_dest = fips
  )
```

New Variables

Several new variables were developed for use in analyzing the provided QVC data.

Rescheduled

The **Rescheduled_Delivery_Dt** variable has a high missing data rate, with 40% of the data observations recorded as NA. The description of this variable indicates that a value would only be recorded if the package was rescheduled, so a shipment that was not rescheduled would be indicated by NA. It was assumed that all of the NA values were shipments that were not rescheduled. Using this assumption, a new binary variable was developed, **Rescheduled**. The binary indicates whether a shipment was rescheduled (1) or was not (0). The **Rescheduled_Delivery_Dt** variable was then dropped.

```
qvc_data <- qvc_data %>%
  mutate(Rescheduled = if_else(is.na(Rescheduled_Delivery_Dt), 0, 1)) %>%
  select(-Rescheduled_Delivery_Dt)
```

Fulfillment_Days

The variable **Fulfillment_Days** is the length of time, in days, between **Order_Dt** and **Delivery_Confirmation_Dt**. The span between **Shipped_Dt** and **Delivery_Confirmation_Dt** was initially considered as well, but after inspection, quality issues regarding the **Shipped_Dt** values lead to the decision not to use **Shipped_Dt**. The span of time between dates was calculated, in part, using the lubridate package (Ref. 3.5).

```
qvc_data <- qvc_data %>%
  mutate(Fulfillment_Days = as.double(difftime(time1 =
ymd(Delivery_Confirmation_Dt)
                                              , time2 = ymd(Order_Dt)
                                              , units = "days"))
         )
```

Distance

The distance between distribution center and shipping destination was calculated for each observation. The location of the distribution center and shipping destination was approximated as the central latitude and longitude of the zip codes of those locations, as provide by the **zip_codes** data set. The geosphere package (3.4) was used to calculate the distance, using the latitude and longitude of the origin and destination points. The Haversine was used. The function distm() returns the distance in meters, so a conversion factor of 0.000621371 miles/meter was included.

The calculation of distance is a computationally taxing process. To minimize the process time of this step on possible future runs of the code, a database of origin and destination zip codes was developed, with the distance calculated between the points.

```
#Zip Code Distance Database
zip_code_dist_database <- qvc_data %>%
  select(zip_distctr:lon_distctr, zip_dest:lon_dest) %>%
  distinct() %>%
  rowwise() %>%
  mutate(Distance = round((distm(x = c(lon_distctr, lat_distctr)
                          , y = c(lon_dest, lat_dest)
                          , fun = distHaversine) * 0.000621371), 0)
  ) %>%
  ungroup() %>%
  select(zip_distctr, zip_dest, Distance)

saveRDS(object = zip_code_dist_database
        , file = "./data_R/output/Group Project Data
set/zip_code_dist_database.RDS")
```

With a database of distances now available, this databased was joined to qvc_data.

```
qvc_data <- qvc_data %>%
  left_join(zip_code_dist_database
            , by = c("zip_distctr",  "zip_dest"))
```

Data Cleaning

Missing Data

At this step in the process the following variables contain missing values:

- **Shipped_Dt**
- **Delivery_Confirmation_Dt**
- **lat_distctr**
- **lon_distctr**
- **lat_dest**
- **lon_dest**
- **Fulfillment_Days**
- **Distance**

None of above variables are have a rate of missing data high enough to imply poor quality of the data, and require dropping the variable. Therefore, the missing values must be imputed or the associated observations dropped.

Delivery_Confirmation_Dt and **Fulfillment_Days** have the largest number of missing values. As **Fulfillment_Days** is a calculated variable based on **Delivery_Confirmation_Dt**, all of the missing values in the former are due to the latter. **Delivery_Confirmation_Dt** has a missing value rate of 4.6%. This missing rate is considered acceptable with such a large data set. These missing values are dropped.

```
qvc_data <- qvc_data %>%
  drop_na(Delivery_Confirmation_Dt)
```

Dropping the **Delivery_Confirmation_Dt** also drops the observations with missing **Shipped_Dt** values.

The only renaming missing values are the missing latitude and longitude values, as well as the corresponding missing **Distance** values. The missing data is due to the **zip_codes** data set being based on zip codes from the 2010 census. This was the most current data documenting all US zip codes, with corresponding geographical data, that could be found. New zip codes have been established since 2010. Distribution centers and shipping destinations from these zip codes is generating the missing values. As the missing **Distance** values are only 0.78% of the data set, these missing values are dropped.

```
qvc_data <- qvc_data %>%
  drop_na(Distance)
```

Data Quality

Inspection of the data shows that observations exist which have a negative **Fulfillment_Days** value. This would me the **Delivery_Confirmation_Dt** pre-dates the **Order_Dt**. This is considered a quality issue with the data. The number of observations with **Fulfillment_Days** less than one day constitutes 0.05% of the data set. All observations with a **Fulfillment_Days** value of less than one are dropped.

Note: All but one instance of a **Fulfillment_Days** value of less than one originated at Distribution Center 0540 (Rocky Mount, NC). There may be a systematic problem related to this location and should be investigated.

```
qvc_data <- qvc_data %>%
  filter(Fulfillment_Days >= 1)
```

With the data processed and cleaned, the resulting base data set **qvc_data** is used as the base input into analyzing the questions posed by this project.

Additional Data

Shape File

For Question 1, one of the visuals created in Tableau was a county choropleth based on total sales in each county within California, Texas, and Florida. While Tableau is able to plot state-based data by default, additional information is required to plot county level data. The county level shape files were provided by the US Census Bureau through the 2018 Tiger/Lines Shape File (Ref. 2.2).

State Information

For additional contextual information, a data set of information on the US states was developed based on available base R data sets, with some modifications.

```
# Develop data frame of state name, abbreviations, regions
state_df <- data.frame(State = state.name
                       , Abb = state.abb
                       , Region = state.region
          )

# Re-define Regions from 4 to 5
Southwest <- c("Arizona", "New Mexico", "Texas", "Oklahoma")

state_df <- state_df %>%
  mutate(Region = as.character(Region),
         Region = if_else(Region == "North Central", "Midwest", Region),
         Region = if_else(Region == "South", "Southeast", Region),
         Region = if_else(State %in% Southwest, "Southwest", Region))

# Washington DC
state_df_DC <- data.frame( State = "Washington, D.C.", Abb = "DC", Region =
"Southeast")

# Puerto Rico
state_df_PR <- data.frame( State = "Puerto Rico", Abb = "PR", Region =
"Southeast")

# Bind Washington DC Dataframe into Main State DF
state_df <- rbind(state_df, state_df_DC, state_df_PR)
```

Question 1

Does the current distribution network maximize customer penetration (spend)? If not, what should QVC do to increase customer penetration with the current distribution network?

The primary analysis for Question 1 was performed via visualization in Tableau. The analysis can be split into two categories: analysis based around US State aggregation, and analysis built around US County aggregation. Both analyses required different data sets.

State Based Analysis

Not all of the variables included in the base data set are required for State level analyses in Question 1. Unnecessary variables are dropped. The following data set was used for all state based Tableau visualizations.

```
qvc_data_Q1_state <- qvc_data %>%
  select(Sales_Order_Nbr,
         Party_Id,
         Total_Line_Amt,
         Merchandise_Dept_Desc,
         Source_Ship_Warehouse_Nbr:Distance,
         -Rescheduled )
```

County Level Visualizations

For county level visualizations, the calculation of total sales by FIPS code was necessary. Currently the QVC data set only has FIPS code information for areas where a purchase was shipped to (**fips_dest**). If a county visualization was done only using these FIPS values, any county where no purchases were shipped to would not be present in the data, producing blank spaces on a map.

In order to avoid blank spaces, a full list of US FIPS codes is joined to a modified the QVC sales data set, ensuring all FIPS code are included in the resultant data set. The only two variables required from the base data set are sales (**Total_Line_Amt**) and the FIPs code for the shipping destination (**fips_dest**).

Once the full FIPS code list is joined to the modified sales data set, all FIPS codes which do not have corresponding sales data set will have NA for **Total_Line_Amt**. All NAs are then replaced by 0, indicating no sales were made in those FIPS regions.

The state of each FIPS code is included in the full FIPS list for to act as a filter for creating visuals in Tableau

```
# Create List of Every5-digit FIPS code
fips_codes_full <- fips_codes %>%
  mutate(fips_code = paste(state_code, county_code, sep = "")) %>%
  distinct() %>%
  select(fips_code, state)
```

```
# Create Data Frame of QVC Sales and FIPS Code
qvc_sales_by_fips <- qvc_data %>%
  select(Total_Line_Amt, fips_dest)

# Join Total FIPS List to Sales Data and Replace NAs with zero
qvc_data_Q1_FIPS <- qvc_sales_by_fips %>%
  right_join(fips_codes_full,
             by = c("fips_dest" = "fips_code")
  ) %>%
  mutate(Total_Line_Amt = coalesce(Total_Line_Amt, 0))
```

Sales / Population Linear Regression

In analyzing the relationship between sales within a state and the state's population, a linear regression analysis was performed. A new data set focused on sales by state was necessary to perform the regression. The data set used to perform the regression needed to include each state's population and total sales.

First, it was required to join the US Census data set to the state information data set. This was to ensure the census data had the correct information to successfully be joined to the sales data. Additionally, the population variable was renamed to **Population_2018** for easier comprehension.

```
us_census_2018 <- us_census_2018 %>%
  left_join(select(state_df, State, Abb)
            , by = c("NAME" = "State")
) %>%
  rename(Population_2018 = POPESTIMATE2018)
```

With the census data prepared, it was then joined to the QVC sales data aggregated by state.

```
qvc_data_stat_pop <- qvc_data %>%
  # Calculate Sales By State
  group_by(state_dest) %>%
  summarise(Sales_Per_State = sum(Total_Line_Amt)
            ) %>%

  #Join State Population Data
left_join(us_census_2018
          , by = c("state_dest" = "Abb")) %>%
  select(-STATE)
```

Linear regression was then performed with the new data, comparing **Sales_Per_State** to **Population_2018**.

```
lm(formula = Sales_Per_State  ~ Population_2018
    , data = qvc_data_stat_pop)

##
## Call:
## lm(formula = Sales_Per_State ~ Population_2018, data = qvc_data_stat_pop)
##
## Coefficients:
##     (Intercept)  Population_2018
##       3.371e+05        4.212e-01
```

Question 2

Are there specific products or product categories that should be located in specific distribution centers?

Analysis of product categories was performed using the base data set, **qvc_data**, in Tableau.

For specific product analysis, Association Rule mining was performed. A new data frame was developed listing only **Sales_Order_Nbr** and **Product_Id** .

```
qvc_data_am <- qvc_data %>%
  select(Sales_Order_Nbr,
         Product_Id)
```

In order to perform Association Mining with the **Arules** package (Ref. 3.1), the input data needs to be in the form of a transactions object, not a standard data frame. To convert the data frame to a transaction object, the data frame is exported as a *.csv* file and then re-imported, using the read.transactions function.

```
write.csv(x = qvc_data_am
        , file = "./data_R/output/q2_association_mining/qvc_data_sales-and-
prod.csv"
        , row.names = FALSE)
qvc_sales_trans <- read.transactions(file =
"./data_R/output/q2_association_mining/qvc_data_sales-and-prod.csv"
                  , format = "single"
                  , cols = c(1,2)
                  , sep = ","
                  , rm.duplicates = TRUE
                  , skip = 1)
```

With the data now in a transactions object form, frequent sets analysis and rules mining analysis are performed.

Frequent Sets

```
# Frequent Sets
frqsets.qvc <- apriori(data = qvc_sales_trans
                       , parameter=list(minlen=2
                                        , supp=1e-5
                                        , conf=0.5
                                        , target="frequent itemsets")
                       , control = list(verbose = FALSE))

# Rules Mining
rules.qvc <- apriori(data = qvc_sales_trans
                     , control = list(verbose=FALSE)
                     , parameter = list(minlen=2
                                        , supp = 1e-7
                                        , conf=0.5))


# Determine and Prune Redundant Rules
redundant <- which (colSums(is.subset(rules.qvc, rules.qvc)) > 1)

rules.qvc.pruned <- rules.qvc[-redundant]
```

Question 3

Do customers that receive their product sooner purchase more than customers with longer delivery times?

A new data set was developed to analyze Question 3. The data set was limited to each customer's first purchase. Each customer was then identified as a repeat or non-repeat customer. Subsequent logistic regression and random forest analysis could then analyze how the applicable variables impacted the chance of being a repeat customer.

First the data frame of initial orders is created.

```
q3_orders_init <- q3_model %>%
  group_by(Party_Id) %>%
  slice(which.min(Order_Dt))
```

The, a data frame identifying which customers are repeat or not is created.

```
qvc_customer_repeat <- qvc_data %>%
  distinct() %>%
  select(Party_Id, Sales_Order_Nbr) %>%
  group_by(Party_Id) %>%
  count() %>%
  mutate(Repeat = if_else(n > 1, 1, 0)) %>%
  select(-n)
```

Finally, the two data frames are joined, yielding the data set q3_model_df. This data set is then used in random forest and logistic regression analyses.

```
q3_model_df <- q3_orders_init %>%
  inner_join(qvc_customer_repeat)
```

The Logistic Regression model was created in Python with the following code:

```
#creation of dummy variables for 'Merchandise_Dept_Desc'
cat_vars=['Merchandise_Dept_Desc']
for var in cat_vars:
   cat_list='var'+'_'+var
   cat_list = pd.get_dummies(updated_data set[var], prefix=var)
   data1=updated_data set.join(cat_list)
   updated_data set=data1

#set of utilized variables for the Logistic Regression model
Inds=updated_data
set.iloc[:,[3,4,9,26,27,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48]]
inds = inds.columns

#Logistic Model creation
X = updated_data set[inds]
y = updated_data set.loc[:, ['Repeat']].values

from sklearn.linear_model import LogisticRegression
from sklearn import metrics
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.25, random_state=5)
logreg = LogisticRegression()
logreg.fit(X_train, y_train)

y_pred = logreg.predict(X_test)
print('Accuracy of logistic regression classifier on test set: {:.2f}'.format(logreg.score(X_test,
y_test)))

from sklearn.metrics import confusion_matrix
confusion_matrix = confusion_matrix(y_test, y_pred)
print(confusion_matrix)

from sklearn.metrics import classification_report
print(classification_report(y_test, y_pred))
```

```
#to create the model summary
import statsmodels.api as sm
logit_model=sm.Logit(y,X)
result=logit_model.fit()
print(result.summary2())

#create ROC curve
from sklearn.metrics import roc_auc_score
from sklearn.metrics import roc_curve
logit_roc_auc = roc_auc_score(y_test, logreg.predict(X_test))
fpr, tpr, thresholds = roc_curve(y_test, logreg.predict_proba(X_test)[:,1])
plt.figure()
plt.plot(fpr, tpr, label='Logistic Regression (area = %0.2f)' % logit_roc_auc)
plt.plot([0, 1], [0, 1],'r--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver operating characteristic')
plt.legend(loc="lower right")
plt.show()
```

The code illustrates how the Random Forest model was created.

```
model_df_trim <- q3_model_df %>%
  ungroup() %>%
  select(-c(Shipped_Dt, Order_Dt, Delivery_Confirmation_Dt,
Sales_Order_Nbr,Party_Id, Package_Id,
            Product_Id, Source_Ship_Warehouse_Nbr, zip_dest, city_distctr,
            city_dest, zip_distctr, fips_dest, lat_distctr, lon_distctr))

model_df_trim <- model_df_trim %>%
  mutate(Line_Status_Cd = factor(Line_Status_Cd),
         state_distctr = factor(state_distctr),
         state_dest = factor(state_dest)
  )



sample = sample.split(model_df_trim$Repeat, SplitRatio = .70)
df.train = subset(model_df_trim, sample == TRUE)
df.test = subset(model_df_trim, sample == FALSE)

rf <- randomForest(as.factor(Repeat)~., data=df.train, importance=TRUE,
ntree=1000, do.trace = 50)

varImpPlot(rf)
```

```
mtry <- tuneRF(df.train[-15], as.factor(df.train$Repeat), ntreeTry=1000,
               stepFactor=0.5, improve=0.01, trace=TRUE, plot=TRUE, do.trace
= TRUE)
best.m <- mtry[mtry[, 2] == min(mtry[, 2]), 1]
rf_tuned <- randomForest(as.factor(Repeat)~., data=df.train, mtry=best.m,
                  importance=TRUE, ntree=1000, do.trace = 50)
varImpPlot(rf_tuned)
pred = predict(rf_tuned, df.test)
print(table(pred, df.test$Repeat))
roc.curve(df.test$Repeat, pred, plotit = TRUE,add =TRUE)
perf <- performance(pred, "tpr", "fpr")
plot(perf)
abline(a=0,b=1)
```

Suggestions

After analyzing the provided data, the following additional information may enhance future analyses:

1. Granular product data. Such data could provide more actionable insights regarding what products to stock at new and existing distribution centers.
2. Detailed customer data. Connecting the classification models predicting repeat business to a customer loyalty program would be useful in strategizing how to push targeted advertising.
3. Additional insights into company practices in data collection. For several merchandise description categories, such as Return, it was unclear on how that category functioned, and if the category should be included in the general model, or perhaps handled separately.
4. Market research on similar companies and the industry in general. Understanding general trends in the industry and competitors' actions would be useful in providing context for our recommendations and developing a more impactful strategy.

References

1. Software

1.1. R Core Team (2019). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

2. Additional Data Input

2.1. United States Census Bureau. (2018). Annual Population Estimates, Estimated Components of Resident Population Change, and Rates of the Components of Resident Population Change for the United States, States, and Puerto Rico: April 1, 2010 to July 1, 2018 (Report No. NST-EST2018). Retrieved from **https://www.census.gov/newsroom/press-kits/2018/pop-estimates-national-state.html**

2.2. 2018 TIGER/Line Shapefiles (machine-readable data files) / prepared by the U.S. Census Bureau, 2018

3. R Packages

3.1. Michael Hahsler, Christian Buchta, Bettina Gruen and Kurt Hornik (2019). arules: Mining Association Rules and Frequent Itemsets. R package version 1.6-3. https://CRAN.R-project.org/package=arules

3.2. Michael Hahsler (2019). arulesViz: Visualizing Association Rules and Frequent Itemsets. R package version 1.3-3. **https://CRAN.R-project.org/package=arulesViz**

3.3. Jarek Tuszynski (2019). caTools: Tools: moving window statistics, GIF, Base64, ROC AUC, etc.. R package version 1.17.1.2. https://CRAN.R-project.org/package=caTools

3.4. Robert J. Hijmans (2019). geosphere: Spherical Trigonometry. R package version 1.5-10. https://CRAN.R-project.org/package=geosphere

3.5. Garrett Grolemund, Hadley Wickham (2011). Dates and Times Made Easy with lubridate. Journal of Statistical Software, 40(3), 1-25. URL **http://www.jstatsoft.org/v40/i03/**.

3.6. John A. Ramey (2014). noncensus: U.S. Census Regional and Demographic Data. R package version 0.1. **https://CRAN.R-project.org/package=noncensus**

3.7. Lionel Henry and Hadley Wickham (2019). purrr: Functional Programming Tools. R package version 0.3.2. **https://CRAN.R-project.org/package=purrr**

3.8. A. Liaw and M. Wiener (2002). Classification and Regression by randomForest. R News 2(3), 18--22.

3.9. Sing T, Sander O, Beerenwinkel N, Lengauer T (2005). “ROCR: visualizing classifier performance in R. ”_Bioinformatics_, *21*(20), 7881. <URL: http://rocr.bioinf.mpi-sb.mpg.de>.

3.10. Hadley Wickham and Jennifer Bryan (2019). readxl: Read Excel Files. R package version 1.3.1. **https://CRAN.R-project.org/package=readxl**

3.11. Nicola Lunardon, Giovanna Menardi, and Nicola Torelli (2014). ROSE: a Package for Binary Imbalanced Learning. R Journal, 6(1), 82-92.

3.12. Kyle Walker (2019). tidycensus: Load US Census Boundary and Attribute Data as 'tidyverse' and 'sf'-Ready Data Frames. R package version 0.9.2. **https://CRAN.R-project.org/package=tidycensus**

3.13. Hadley Wickham (2017). tidyverse: Easily Install and Load the 'Tidyverse'. R package version 1.2.1. https://CRAN.R-project.org/package=tidyverse

3.14. Hadley Wickham (2019). stringr: Simple, Consistent Wrappers for Common String Operations. R package version 1.4.0. https://CRAN.R-project.org/package=stringr