Data Mining of Ecommerce Dataset (Problem Solving Based on Experience)

Ibrahim Sanusi

2023-06-25

Loading necessary packages

Loading the data sets

The data set was obtained from “https://www.kaggle.com/datasets/olistbr/brazilian-ecommerce (https://www.kaggle.com/datasets/olistbr/brazilian-ecommerce)” and consist of 8 files in the following categories: Customer information, Order location, Items purchased, Payment descriptions, Order processing, Product categories, Product descriptions, and Sellers information.

Data Cleaning

- Check for missing data
- Check for data irregularities

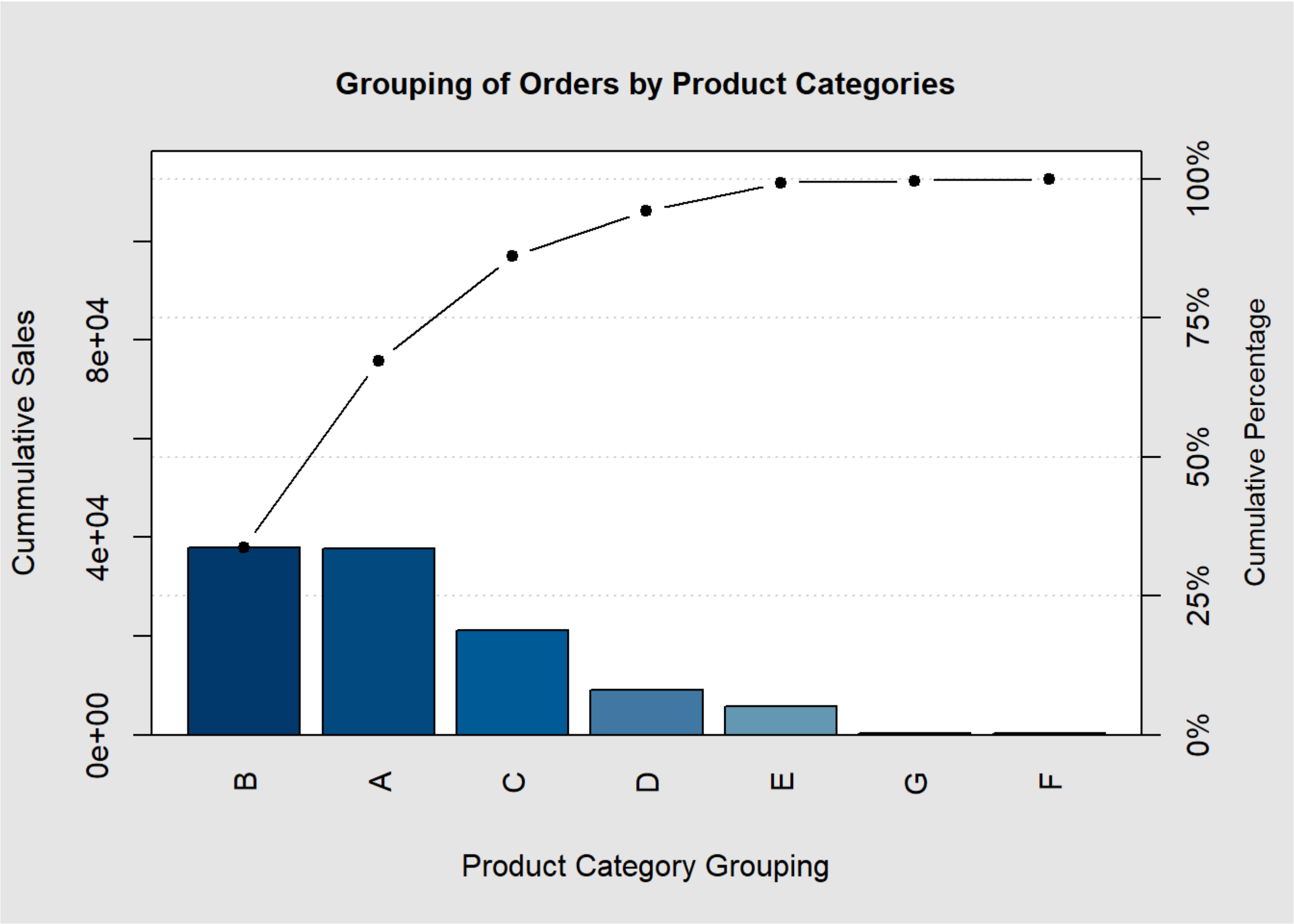
The Product data set has 838 missing values), Product[66008,] can be used to check out one of the row.The missing values will be handled in future

This project aim to solve the following problems:

1. Arrange products in the FC/DC based on volumes of order in product categories
   Functions Used: Select + Merge + Group + Summarize + Arrange + Rename + ifelse + Mutate + Paretoplot

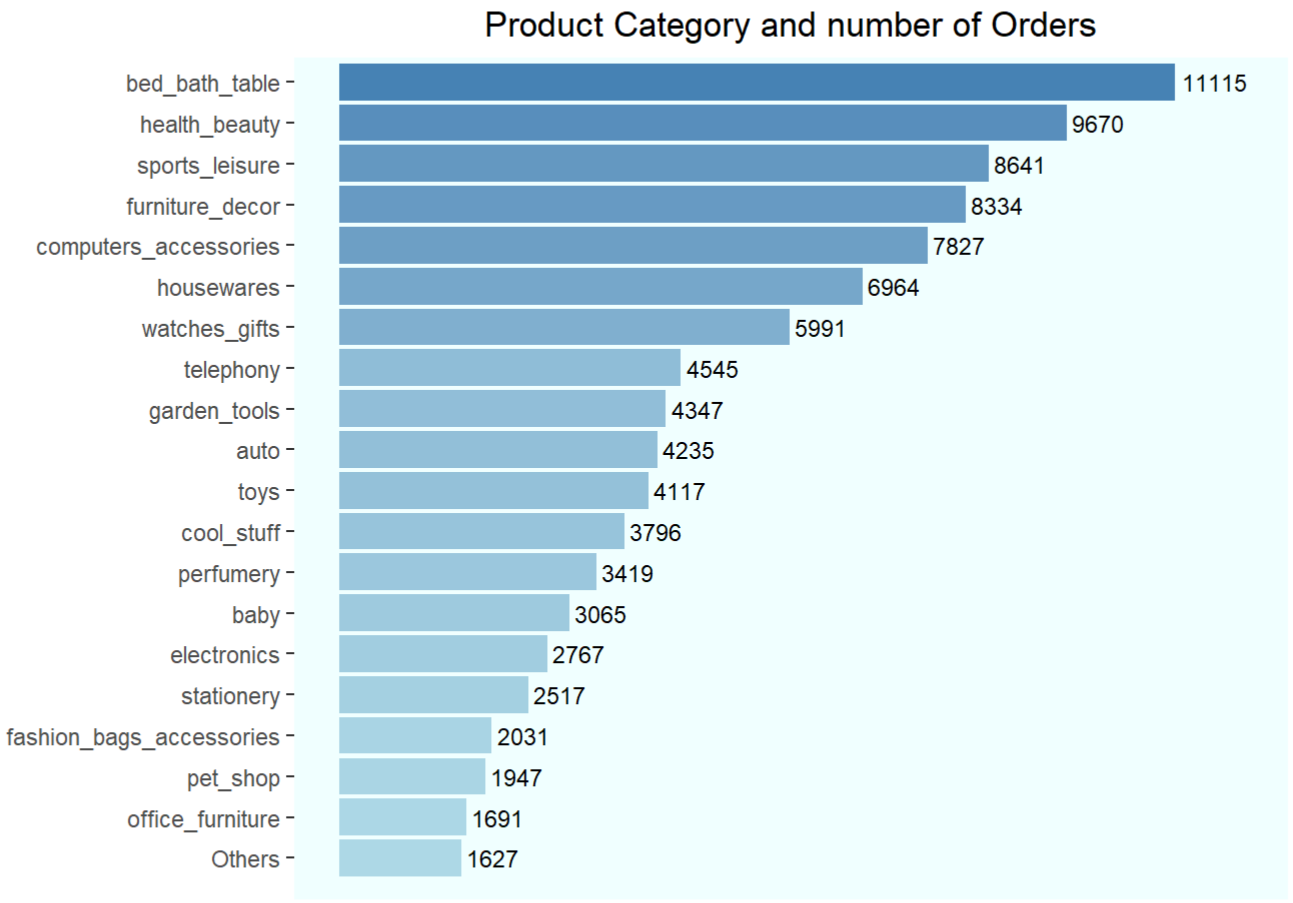
```
Items1 <- subset (Items, select = c('order_id', 'product_id'))
Product1 <- subset (Product, select = c('product_id', 'product_category_name'))
ProductOrdered <-merge(Items1,Product1,
                        by.x="product_id", all = TRUE)
ProductOrdered2 <-merge(ProductOrdered,ProductCategory,
                        by.x = "product_category_name", by.y = "ï..product_category_name", all =
TRUE)
ProductOrderedFrequency <- ProductOrdered2%>%
  group_by(product_category_name_english) %>%
  summarize(Freq=n())%>%
  arrange(desc(Freq))
ProductOrderedFrequency[20,1] <- 'Others'
Paretoplot <- ProductOrderedFrequency %>%
  mutate(Pareto = ifelse(Freq > 8000, 'A',
                         (ifelse(Freq > 4000, 'B',
                                 (ifelse(Freq > 1627, 'C',
                                         ifelse(Freq > 500, 'D',
                                                ifelse(Freq > 100, 'E',
                                                       ifelse(Freq > 50, 'F',
                                        ifelse(Freq > 1, 'G', 'H'))))))))))
Paretoplot2 <- Paretoplot %>%
  mutate(Percentage = round(cumsum(100*Freq/sum(Freq))))
Paretoplot3 <- Paretoplot2 %>%
  group_by(Pareto) %>%
  summarise(Pareto_Analysis = sum(`Freq`))%>%
  mutate(Percentage = round(cumsum(100*Pareto_Analysis/sum(Pareto_Analysis))))

pareto.chart(Paretoplot3$Pareto_Analysis, main = "Grouping of Orders by Product Categories", xla
b = "Product Category Grouping", ylab = "Cummulative Sales")
```

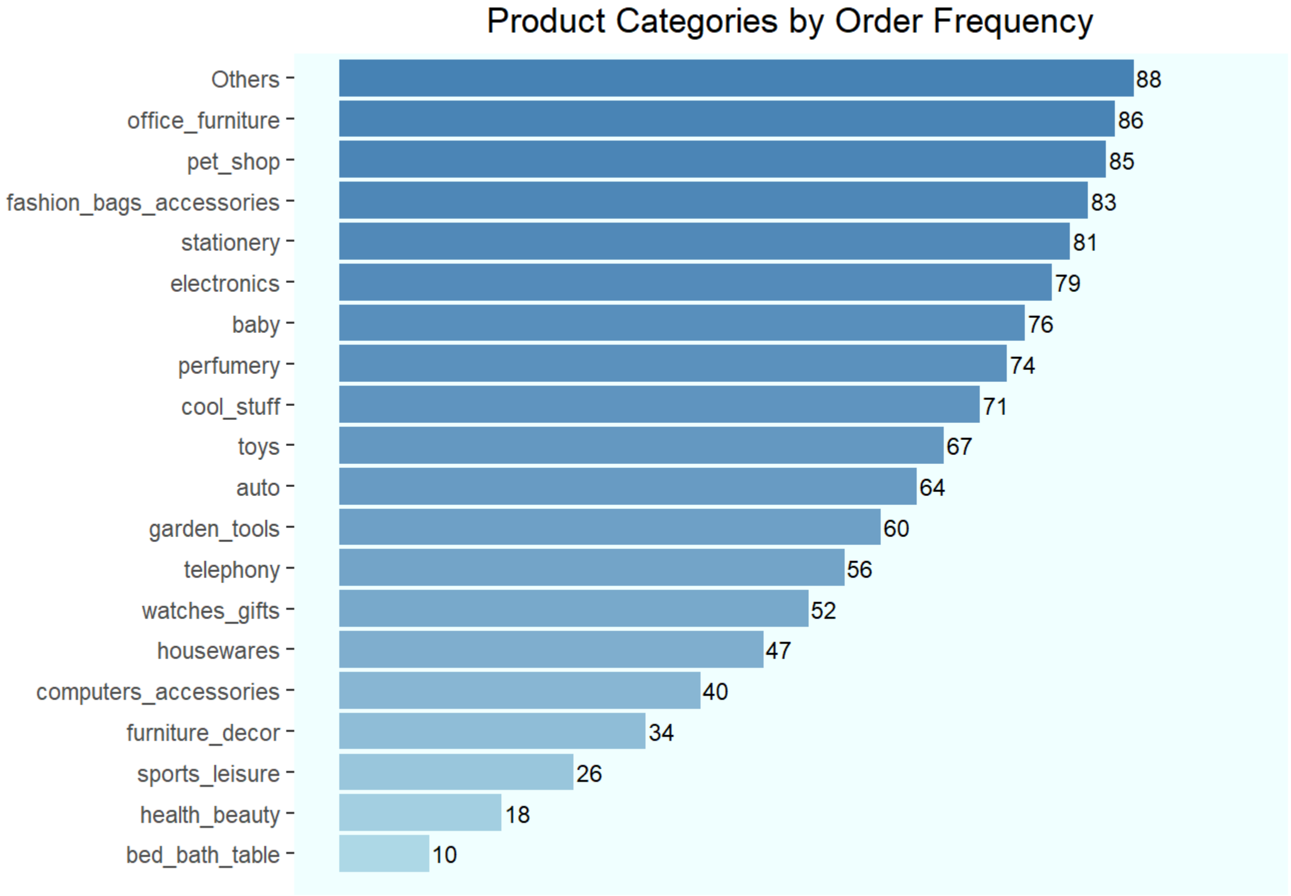


```
##
## Pareto chart analysis for Paretoplot3$Pareto_Analysis
##        Frequency    Cum.Freq.   Percentage Cum.Percent.
##   B 3.802600e+04 3.802600e+04 3.375588e+01 3.375588e+01
##   A 3.776000e+04 7.578600e+04 3.351975e+01 6.727563e+01
##   C 2.123300e+04 9.701900e+04 1.884865e+01 8.612428e+01
##   D 9.085000e+03 1.061040e+05 8.064802e+00 9.418908e+01
##   E 5.830000e+03 1.119340e+05 5.175322e+00 9.936440e+01
##   G 3.610000e+02 1.122950e+05 3.204616e-01 9.968486e+01
##   F 3.550000e+02 1.126500e+05 3.151354e-01 1.000000e+02
```

```
Paretoplot2 %>%
  top_n(20, Freq) %>%
  ggplot(aes(x=Freq,y=reorder(product_category_name_english,Freq), fill=(Freq)))+
  geom_col()+
  geom_text(aes(label = Freq), vjust = 0.5, hjust=-0.1, size=3)+
  expand_limits(x = 12000, y = 0)+
  scale_fill_gradient(low="lightblue", high = "steelblue")+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title="", x="", y="")+
  ggtitle("Product Category and number of Orders") +
  theme(legend.position = (""))+
  theme(plot.title = element_text(hjust = 0.5))+
  theme(axis.text.x = element_blank(),
        axis.ticks.x = element_blank(),
        panel.background = element_rect(fill = "azure", color="azure"))
```


Product Category and number of Orders

```
Paretoplot2 %>%
  top_n(20, -Percentage) %>%
  ggplot(aes(x=Percentage,y=reorder(product_category_name_english,Percentage), fill=(Percentag
e)))+
  geom_col()+
  geom_text(aes(label = Percentage), vjust = 0.5, hjust=-0.1, size=3)+
  expand_limits(x = 100, y = 0)+
  scale_fill_gradient(low="lightblue", high = "steelblue")+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title="", x="", y="")+
  ggtitle("Product Categories by Order Frequency") +
  theme(legend.position = (""))+
  theme(plot.title = element_text(hjust = 0.5))+
  theme(axis.text.x = element_blank(),
        axis.ticks.x = element_blank(),
        panel.background = element_rect(fill = "azure", color="azure"))
```



Based on “Grouping of Orders by Product Categories” and “Product Categories by Order Frequency” figures above, products in categories A, B, and C or first 16 product categories (i.e. ‘bed_bath_table’ to ‘stationery’) should be staged very close to the packing and shipping area of the DC/FC as they have the highest volume of orders (by doing this alone, the FC/DC will solve 80% of their order processing efficiency).

2. Solve safety problem base of product descriptions
3. Explain trend base on days of the week, month, days and hourly to solve staffing problem
4. Describe order processing efficiency (same day, on-time, late ship, late delivery)
5. Rank seller base on order processing and sales
6. Payment Analysis (credit card, installment)
7. Does items description length influence sales or purchases
8. Create a dashboard with your data

NOTE: This is an ongoing project and I will update as I solve each of the problems