TidyVerse Part 2

Priya Shaji

5/3/2019

## Tidyverse Assignment Part 2

Expansion of Santosh Cheruku's Black friday sales dataset.

Santosh Cheruku’s exloratory data analysis and code on Black Friday sales can be found on the following github link:

<https://github.com/acatlin/SPRING2019TIDYVERSE/blob/master/Tidyverse_Mutate_functions.rmd>

The dataset here is a sample of the transactions made in a retail store. The store wants to know better the customer purchase behaviour against different products. Specifically, here the problem is a regression problem where we are trying to predict the dependent variable (the amount of purchase) with the help of the information contained in the other variables.

Classification problem can also be settled in this dataset since several variables are categorical, and some other approaches could be “Predicting the age of the consumer” or even “Predict the category of goods bought”. This dataset is also particularly convenient for clustering and maybe find different clusters of consumers within it.

In this part 2 , I will focus on data clustering and grouping of the black friday sales by age and purchase of goods.

### Load Data

Fistly, we will load data and libraries

library(tidyverse)

## ── Attaching packages ─────────────────────────────────────────────────────── tidyverse 1.2.1 ──

## ✔ ggplot2 3.1.1 ✔ purrr 0.3.2   
## ✔ tibble 2.1.1 ✔ dplyr 0.8.0.1  
## ✔ tidyr 0.8.3 ✔ stringr 1.4.0   
## ✔ readr 1.3.1 ✔ forcats 0.4.0

## ── Conflicts ────────────────────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

library(DT)

Load the dataset

sales\_df <- read.csv("https://raw.githubusercontent.com/san123i/CUNY/master/Semester1/607/Tidyverse\_assignment\_data/BlackFriday.csv")  
head(sales\_df)

## User\_ID Product\_ID Gender Age Purchase  
## 1 1000001 P00069042 F 0-17 8370  
## 2 1000001 P00248942 F 0-17 15200  
## 3 1000001 P00087842 F 0-17 1422  
## 4 1000001 P00085442 F 0-17 1057  
## 5 1000002 P00285442 M 55+ 7969  
## 6 1000003 P00193542 M 26-35 15227

### Exploratory Data Analysis

Let’s see total purchasers on black friday

# total purchaser  
sales\_df %>%  
distinct(User\_ID) %>%  
nrow() %>%  
paste("buyers registered at Black Friday")

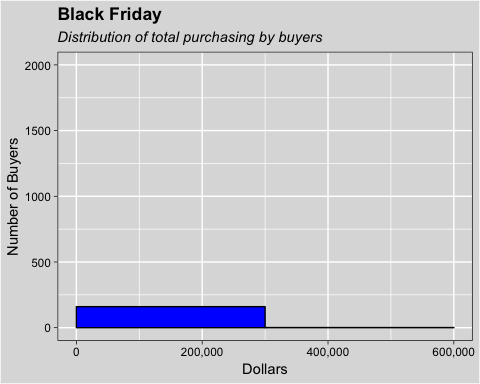
## [1] "161 buyers registered at Black Friday"

### Groups and Distributions

#### Total Purchase Distribution

Here the distribution of goods will be analyzed based on purchase.

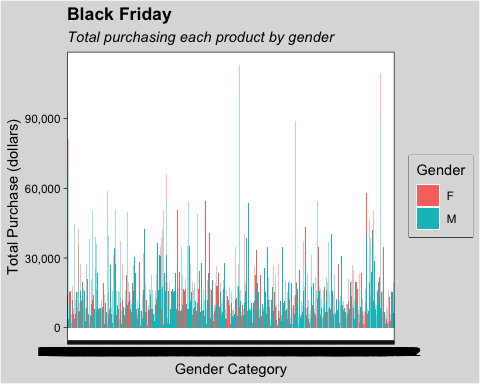
#total purchase distribution  
sales\_df %>%  
group\_by(User\_ID) %>%  
summarise(total\_purchase = sum(Purchase)) %>%  
ggplot(aes(x = total\_purchase)) +   
geom\_histogram(col = 'black', fill = 'blue', binwidth = 300000, center = 150000) +  
theme\_linedraw() +   
theme(panel.background = element\_rect(fill = "gainsboro", colour = "white", size = 0.5, linetype = "solid"), #theme panel settings  
 plot.background = element\_rect(fill = "gainsboro"), #theme panel settings  
 panel.grid.major = element\_line(size = 0.5, linetype = 'solid', colour = "white"), #theme panel settings  
 panel.grid.minor = element\_line(size = 0.25, linetype = 'solid', colour = "white"), #theme panel settings  
 plot.title = element\_text(hjust = 0, face = 'bold',color = 'black'), #title settings  
 plot.subtitle = element\_text(face = "italic")) + #subtitle settings  
labs(x = 'Dollars', y = 'Number of Buyers', title = "Black Friday", #name title and axis  
 subtitle = "Distribution of total purchasing by buyers") + #name subtitle  
scale\_y\_continuous(limits = c(0,2000), breaks = c(0,500,1000,1500,2000)) + #set axis limits and break  
scale\_x\_continuous(labels = scales::comma) #prevent scientific number in x-axis



#### Total Purchase by Product ID and Gender

Here the distribution of goods will be analyzed based on Product ID and Gender.

#total purchase distribution grouping by City and Gender  
sales\_df %>%  
group\_by(Product\_ID, Gender) %>%  
summarise(total\_purchase = sum(Purchase)) %>%  
ggplot(aes(x=Product\_ID, y = total\_purchase)) +  
geom\_col(aes(fill = Gender)) +  
theme\_linedraw() +   
theme(legend.box.background = element\_rect(colour = "black"),  
 legend.background = element\_rect(fill = "gainsboro"),  
 panel.background = element\_rect(fill = "gainsboro", colour = "white", size = 0.5, linetype = "solid"), #theme panel settings  
 plot.background = element\_rect(fill = "gainsboro"), #theme panel settings  
 panel.grid.major = element\_line(size = 0.5, linetype = 'solid', colour = "white"), #theme panel settings  
 panel.grid.minor = element\_line(size = 0.25, linetype = 'solid', colour = "white"), #theme panel settings  
 plot.title = element\_text(hjust = 0, face = 'bold',color = 'black'), #title settings  
 plot.subtitle = element\_text(face = "italic")) + #subtitle settings  
labs(x = 'Gender Category', y = 'Total Purchase (dollars)', title = "Black Friday", #name title and axis  
 subtitle = "Total purchasing each product by gender") + #name subtitle  
guides(fill=guide\_legend(title = "Gender")) + #remove color legend  
scale\_y\_continuous(labels = scales::comma) #prevent scientific number in x-axis



#### Total Purchase by Gender

Here the distribution of goods will be analyzed based on Gender.

#gender  
gender <- sales\_df %>%  
 group\_by(Gender) %>%  
 distinct(User\_ID) %>%  
 summarise(Total=n())  
gender

## # A tibble: 2 x 2  
## Gender Total  
## <fct> <int>  
## 1 F 49  
## 2 M 112

#### Total Purchase by Age

Here the distribution of goods will be analyzed based on Age.

#Age  
age <- sales\_df %>%  
 group\_by(Age) %>%  
 summarise(Total=n())  
age

## # A tibble: 7 x 2  
## Age Total  
## <fct> <int>  
## 1 0-17 42  
## 2 18-25 214  
## 3 26-35 342  
## 4 36-45 200  
## 5 46-50 121  
## 6 51-55 54  
## 7 55+ 28

### Data Clustering

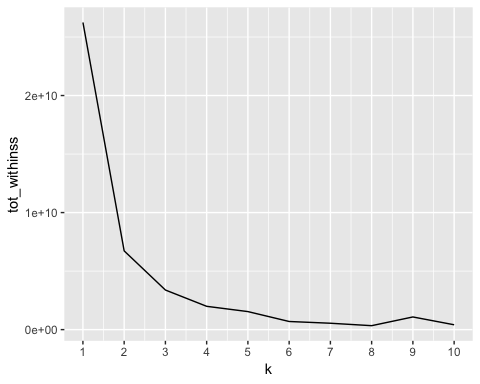
Clustering is a broad set of techniques for finding subgroups of observations within a data set. When we cluster observations, we want observations in the same group to be similar and observations in different groups to be dissimilar. Because there isn’t a response variable, this is an unsupervised method, which implies that it seeks to find relationships between the n observations without being trained by a response variable.

BlackFridayForClustering <- sales\_df %>%  
 select(Purchase)

#### Number of Cluster

Determine the Number of Cluster

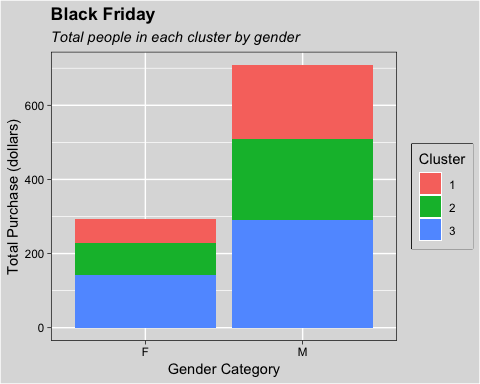
# Use map\_dbl to run many models with varying value of k (centers)  
tot\_withinss <- map\_dbl(1:10, function(k){  
 model <- kmeans(x = BlackFridayForClustering, centers = k)  
 model$tot.withinss  
})  
  
# Generate a data frame containing both k and tot\_withinss  
elbow\_df <- data.frame(  
 k = 1:10,  
 tot\_withinss = tot\_withinss  
)  
  
# Plot the elbow plot  
ggplot(elbow\_df, aes(x = k, y = tot\_withinss)) +  
 geom\_line() +  
 scale\_x\_continuous(breaks = 1:10)



#### Cluster Model

# Build a kmeans model  
model\_km3 <- kmeans(BlackFridayForClustering, centers = 3)  
  
# Extract the cluster assignment vector from the kmeans model  
clust\_km3 <- model\_km3$cluster  
  
# Create a new dataframe appending the cluster assignment  
BlackFriday\_Clust <- mutate(sales\_df, cluster = clust\_km3)  
  
# summarise the clustering  
BlackFriday\_Clust\_Note <- BlackFriday\_Clust %>%  
 group\_by(cluster) %>%  
 summarise(min\_purchase = min(Purchase),  
 max\_purchase = max(Purchase),  
 avg\_purchase = round(mean(Purchase),0))

# how many people in each cluster  
BlackFriday\_Clust %>%  
group\_by(Gender, cluster) %>%  
summarise(n = n()) %>%  
ggplot(aes(x=Gender, y = n)) +  
geom\_col(aes(fill = as.factor(cluster))) +  
theme\_linedraw() +   
theme(legend.box.background = element\_rect(colour = "black"),  
 legend.background = element\_rect(fill = "gainsboro"),  
 panel.background = element\_rect(fill = "gainsboro", colour = "white", size = 0.5, linetype = "solid"), #theme panel settings  
 plot.background = element\_rect(fill = "gainsboro"), #theme panel settings  
 panel.grid.major = element\_line(size = 0.5, linetype = 'solid', colour = "white"), #theme panel settings  
 panel.grid.minor = element\_line(size = 0.25, linetype = 'solid', colour = "white"), #theme panel settings  
 plot.title = element\_text(hjust = 0, face = 'bold',color = 'black'), #title settings  
 plot.subtitle = element\_text(face = "italic")) + #subtitle settings  
labs(x = 'Gender Category', y = 'Total Purchase (dollars)', title = "Black Friday", #name title and axis  
 subtitle = "Total people in each cluster by gender") + #name subtitle  
guides(fill=guide\_legend(title = "Cluster")) + #remove color legend  
scale\_y\_continuous(labels = scales::comma) #prevent scientific number in x-axis



### Reference

1. <https://uc-r.github.io/kmeans_clustering>