Exercise 6

Marc Dotson

Let’s return to our recency, frequency, monetary (RFM) analysis. Now we can analyze customers based on how much they have spent. Once again, *remember to sketch out what you’d like the data to look like.* In RStudio on Posit Cloud, create a new Quarto document and do the following.

1. Let’s define customers who have “high monetary value” as anyone who has spent more than $25,000 total. In its own section of the report, use the store\_revenue data to identify these customers. How many “high monetary value” customers are there?
2. In its own section of the report, report on the composition of these customers by visualizing the relationship between their total revenue, credit, gender, and marital status all in a single visualization. Report on what you discover.
3. Render the Quarto document into Word, export the Word document, and upload to Canvas.

**Five points total, one point each for:**

* **Correctly identifying 1,501 “high monetary value” customers.**
* **Question 1 and Question 2 having their own sections.**
* **Producing a visualization that incorporates all four variables.**
* **Interpreting the visualization.**
* **Submitting a rendered Word document.**

## Question 1

Let’s first import store\_revenue. Let’s start with how we retrieve it from the database.

# Load packages.  
library(tidyverse)

── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
✔ dplyr 1.1.4 ✔ readr 2.1.4  
✔ forcats 1.0.0 ✔ stringr 1.5.1  
✔ ggplot2 3.4.4 ✔ tibble 3.2.1  
✔ lubridate 1.9.3 ✔ tidyr 1.3.0  
✔ purrr 1.0.2   
── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
✖ dplyr::filter() masks stats::filter()  
✖ dplyr::lag() masks stats::lag()  
ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(dbplyr)

Attaching package: 'dbplyr'  
  
The following objects are masked from 'package:dplyr':  
  
 ident, sql

library(DBI)

# Connect to the database.  
con <- dbConnect(  
 RPostgreSQL::PostgreSQL(),  
 dbname = "analyticsdb",  
 host = "analyticsdb.ccutuqssh92k.us-west-2.rds.amazonaws.com",  
 port = 55432,  
 user = "quantmktg",  
 password = rstudioapi::askForPassword("Database password")  
)  
  
# Import the data.  
store\_revenue <- tbl(con, "store\_revenue") |>  
 collect()  
  
# Disconnect from the database.  
dbDisconnect(con)  
  
# Write the data locally.  
write\_csv(store\_revenue, here::here("Data", "store\_revenue.csv"))

Or, if you’ve already written it locally.

# Import the data.  
store\_revenue <- read\_csv(here::here("Data", "store\_revenue.csv"))

Rows: 10531 Columns: 169  
── Column specification ────────────────────────────────────────────────────────  
Delimiter: ","  
dbl (169): customer\_id, jan\_2005, feb\_2005, mar\_2005, apr\_2005, may\_2005, ju...  
  
ℹ Use `spec()` to retrieve the full column specification for this data.  
ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

Now we can identify “high monetary value” customers as anyone who has spent more than $25,000 total.

# Tidy and summarize revenue history and filter on respondents  
# who have spent more than $25,000 total.  
money\_cust <- store\_revenue |>   
 pivot\_longer(  
 -customer\_id,  
 names\_to = "month\_year",  
 values\_to = "revenue"  
 ) |>   
 group\_by(customer\_id) |>   
 summarize(total\_revenue = sum(revenue)) |>   
 filter(total\_revenue > 25000)  
   
money\_cust

# A tibble: 1,501 × 2  
 customer\_id total\_revenue  
 <dbl> <dbl>  
 1 1001 25199.  
 2 1011 28604.  
 3 1014 27725.  
 4 1019 37824.  
 5 1020 31032.  
 6 1046 26492.  
 7 1055 26372.  
 8 1056 27140.  
 9 1059 25308.  
10 1071 25694.  
# ℹ 1,491 more rows

As we can see, there are 1,501 customers in our CRM database who qualify as “high monetary value.”

## Question 2

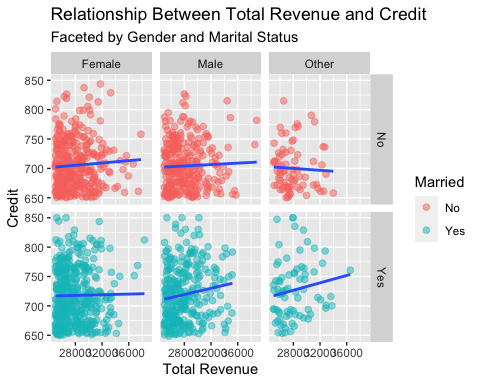
Now we can report on the composition of these customers by visualizing the relationship between their total revenue, credit, gender, and marital status all in a single visualization.

# Import customer data.  
customer\_data <- read\_csv(here::here("Data", "customer\_data.csv"))

Rows: 10531 Columns: 13  
── Column specification ────────────────────────────────────────────────────────  
Delimiter: ","  
chr (8): gender, married, college\_degree, region, state, review\_time, review...  
dbl (5): customer\_id, birth\_year, income, credit, star\_rating  
  
ℹ Use `spec()` to retrieve the full column specification for this data.  
ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

# Join money\_cust and customer\_data and visualize the relationship  
# between total revenue, credit, gender, and marital status.  
money\_cust |>   
 left\_join(customer\_data, join\_by(customer\_id)) |>   
 ggplot(aes(x = total\_revenue, y = credit)) +  
 geom\_point(size = 2, alpha = 0.5 , aes(color = married)) +  
 geom\_smooth(method = "lm", se = FALSE) +  
 facet\_grid(married ~ gender) +  
 labs(  
 title = "Relationship Between Total Revenue and Credit",  
 subtitle = "Faceted by Gender and Marital Status",  
 x = "Total Revenue",  
 y = "Credit"  
 ) +  
 scale\_color\_discrete(name = "Married")

`geom\_smooth()` using formula = 'y ~ x'



It appears that the relationship between total revenue (i.e., how much a customer has spent) and their credit is strongest for unmarried women, married men, and married individuals who have selected “other” as their gender.