Week 3 Challenge

Amara Diallo

12/30/2018

Contents

Challenge Summary

This is a short challenge to begin applying what you are learning to the problem at hand. You will go through a series of questions related to the course project goals:

1. Coming up with a new product idea, and
2. Segmenting the customer-base

Objectives

1. Apply `lubridate` and `stringr` functions to answer questions related to the course projects.
2. Gain exposure to `rmarkdown`.

Data

To read the data, make sure that the paths point to the appropriate data sets. Saving the file in the "challenges folder" should enable the paths to be detected correctly.

```
# Load libraries
library(tidyverse)
library(lubridate)
```

```
# Read bike orderlines data
path_bike_orderlines <- "../00_data/bike_sales/data_wrangled/bike_orderlines.rds"
bike_orderlines_tbl <- read_rds(path_bike_orderlines) %>%

    # Fixing typos found in Feature Engineering
    mutate(model = case_when(
        model == "CAAD Disc Ultegra" ~ "CAAD12 Disc Ultegra",
        model == "Syapse Carbon Tiagra" ~ "Synapse Carbon Tiagra",
        model == "Supersix Evo Hi-Mod Utegra" ~ "Supersix Evo Hi-Mod Ultegra",
        TRUE ~ model
    ))

glimpse(bike_orderlines_tbl)
```

```
## Rows: 15,644
## Columns: 13
## $ order_date     <dttm> 2011-01-07, 2011-01-07, 2011-01-10, 2011-01-10, 201...
## $ order_id       <dbl> 1, 1, 2, 2, 3, 3, 3, 3, 3, 4, 5, 5, 5, 5, 6, 6, 6, 6...
## $ order_line     <dbl> 1, 2, 1, 2, 1, 2, 3, 4, 5, 1, 1, 2, 3, 4, 1, 2, 3, 4...
## $ quantity       <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1...
## $ price          <dbl> 6070, 5970, 2770, 5970, 10660, 3200, 12790, 5330, 15...
## $ total_price    <dbl> 6070, 5970, 2770, 5970, 10660, 3200, 12790, 5330, 15...
## $ model          <chr> "Jekyll Carbon 2", "Trigger Carbon 2", "Beast of the...
## $ category_1     <chr> "Mountain", "Mountain", "Mountain", "Mountain", "Roa...
## $ category_2     <chr> "Over Mountain", "Over Mountain", "Trail", "Over Mou...
## $ frame_material <chr> "Carbon", "Carbon", "Aluminum", "Carbon", "Carbon", ...
## $ bikeshop_name  <chr> "Ithaca Mountain Climbers", "Ithaca Mountain Climber...
## $ city           <chr> "Ithaca", "Ithaca", "Kansas City", "Kansas City", "L...
## $ state          <chr> "NY", "NY", "KS", "KS", "KY", "KY", "KY", "KY", "KY"...
```

Questions

lubridate: Which month has the highest bike sales? (Difficulty = Medium)

- Start with `bike_orderlines_tbl`
- Select columns `order_date` and `total_price`
- Add a column called `month`
- Group by, summarize, and ungroup calculating the `sales`
- Arrange the sales values by month (Jan - Dec)
- Format the sales values as `dollar()`

- Adjust column names to title case

What does this tell us about a time of year to focus marketing efforts?

```
bike_orderlines_tbl %>%
    select(order_date, total_price) %>%
    mutate(month = month(order_date,label = TRUE)) %>%

    group_by(month) %>%
    summarize(sales = sum(total_price)) %>%
    ungroup() %>%

    arrange(month) %>%
    mutate(sales = scales::dollar(sales)) %>%
    set_names(names(.) %>% str_to_title())
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 12 x 2
##    Month Sales
##    <ord> <chr>
##  1 Jan   $4,089,460
##  2 Feb   $5,343,295
##  3 Mar   $7,282,280
##  4 Apr   $8,386,170
##  5 May   $7,935,055
##  6 Jun   $7,813,105
##  7 Jul   $7,602,005
##  8 Aug   $5,346,125
##  9 Sep   $5,556,055
## 10 Oct   $4,394,300
## 11 Nov   $4,169,755
## 12 Dec   $3,114,725
```

stringr: What is the median orderline sales value by Bike Attribute? (Difficulty = Medium)

- Begin with `bike_orderlines`
- Select `model` and `total_price`
- Detect if string is present (e.g. "black inc")
- Groupby, summarize, and ungroup calculating the `median()` orderline
- Format numeric price as `dollar()` (Hint: investigate `largest_with_cents` argument)
- Rename column to evaluation string (e.g. "Black Inc")

Evaluate "Black Inc". *What does this tell us about the "Black Inc" feature?*

```
bike_orderlines_tbl %>%
    select(model, total_price) %>%
    mutate(
        `black inc` = model %>% str_to_lower() %>% str_detect("black") %>% as.numeric(),
```

```
    ) %>%

    group_by(`black inc`) %>%
    summarise(median = median(total_price)) %>%
    ungroup() %>%

    mutate(
        price = scales::dollar(median)
    ) %>%
    set_names(names(.) %>% str_to_title())
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 2 x 3
##   `Black Inc` Median Price
##         <dbl>  <dbl> <chr>
## 1           0   2880 $2,880
## 2           1  12250 $12,250
```

Evaluate "Ultegra". *What does this tell us about the "Ultegra" feature?*

```
bike_orderlines_tbl %>%
    select(model, total_price) %>%
    mutate(
        Ultegra= model %>% str_to_title() %>% str_detect("Ultegra") %>% as.numeric(),
    ) %>%

    group_by(Ultegra) %>%
    summarise(median = median(total_price)) %>%
    ungroup() %>%

    mutate(
        price = scales::dollar(median)
    ) %>%
    set_names(names(.) %>% str_to_upper())
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 2 x 3
##   ULTEGRA MEDIAN PRICE
##     <dbl>  <dbl> <chr>
## 1       0   3200 $3,200
## 2       1   3200 $3,200
```

Evaluate "Disc" option. *What does this tell us about the "Disc" feature?*

```
bike_orderlines_tbl %>%
    select(model, total_price) %>%
    mutate(

        Disc   = model %>% str_to_upper() %>% str_detect("DISC") %>% as.numeric()
```

```
    ) %>%

    group_by(Disc) %>%
    summarise(median = median(total_price)) %>%
    ungroup() %>%

    mutate(
        price = scales::dollar(median)
    ) %>%
    set_names(names(.) %>% str_to_lower())
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 2 x 3
##    disc median price
##   <dbl>  <dbl> <chr>
## 1     0   3200 $3,200
## 2     1   2660 $2,660
```

stringr: What are the average, min, and max prices by Base Model? (Difficulty = High)

- Start with `bike_orderlines_tbl`
- Select distinct primary category, secondary category, model, and price (unit price, not total price)
- Create the base feature, `model_base` (Hint: Use the Feature Engineering code)
  - separate the models
  - Create a base feature that combines the appopriate parts (e.g. "Beast of the East")
- Remove any unnecessary columns (Hint: Deselect any columns matching `"model_[0-9]"`)
- Group by, summarize, and ungroup (Hint: use `mean()`, `min()`, and `max()`)
- Arrange descending by average price
- Format any numeric columns as `dollar()` (Hint: Check out `largest_with_cents`)
- Adjust the column names to title case

What does this tell us about how bikes are priced?

```
bike_orderlines_tbl %>%
    select(category_1, category_2, model, price) %>% distinct() %>%
    separate(col = model ,
             into = str_c("model_", 1:7),
             sep = " ",
             fill = "right",
             remove = FALSE,
             extra="drop") %>%
    mutate(
        model_base = case_when(
            #Fix beast of the east
            str_detect(str_to_lower(model_1),"beast")~str_c(model_1,model_2,model_3,model_4,sep=" "),
            #Fix Supersix Evo
            str_detect(str_to_lower(model_1), "supersix") ~ str_c(model_1, model_2, sep = " "),
```

```
            # Fix Fat CAAD bikes
            str_detect(str_to_lower(model_1), "fat") ~ str_c(model_1, model_2, sep = " "),

            # Fix Bad Habi
            str_detect(str_to_lower(model_1), "bad") ~ str_c(model_1, model_2, sep = " "),

            # Fix Scalpel 29
            str_detect(str_to_lower(model_1), "29") ~ str_c(model_1, model_2, sep = " "),

            # Fix Carbon 2
            str_detect(str_to_lower(model_2), "carbon") ~ str_c(model_2, model_3, sep = " "),

            # Fix hi-mod & rival
            str_detect(str_to_lower(model_2), c("hi-mod","rival") )~ str_c(model_2, model_3, sep=" "),
            # Fix "jekyll"
            str_detect(str_to_lower(model_1), "jekyll") ~ str_c(model_2, model_3, sep = " "),

            TRUE ~ model_1
        )) %>%

    #mutate(model_tier = model %>% str_replace(model_base, replacement = " ") %>% str_trim()) %>%
    select(-matches("model_[0-9]")) %>%

    group_by(category_1, category_2, model_base) %>%
    summarise(avg_price  = mean(price),
              max_price  = max(price),
              min_price  = min(price)) %>%

    arrange(desc(avg_price)) %>%
    mutate_if(is.numeric, scales::dollar) %>%
    set_names(names(.) %>% str_to_title())
```

```
## Warning: Problem with `mutate()` input `model_base`.
## i longer object length is not a multiple of shorter object length
## i Input `model_base` is `case_when(...)`.

## Warning in stri_detect_regex(string, pattern, negate = negate, opts_regex =
## opts(pattern)): longer object length is not a multiple of shorter object length

## `summarise()` regrouping output by 'category_1', 'category_2' (override with `.groups` argument)

## `mutate_if()` ignored the following grouping variables:
## Columns `category_1`, `category_2`

## # A tibble: 36 x 6
## # Groups:   Category_1, Category_2 [9]
##    Category_1 Category_2         Model_base   Avg_price Max_price Min_price
##    <chr>      <chr>              <chr>        <chr>     <chr>     <chr>
##  1 Mountain   Cross Country Race Hi-Mod Team  $9,060.00 $9,060    $9,060
##  2 Mountain   Cross Country Race Scalpel-Si   $8,127.50 $12,790   $3,200
##  3 Mountain   Over Mountain      Carbon 1     $8,095    $8,200    $7,990
##  4 Mountain   Trail              Carbon 1     $7,460.00 $7,460    $7,460
```

```
##  5 Road       Endurance Road     Hi-Mod Disc  $7,460.00 $9,590    $5,330
##  6 Mountain   Cross Country Race Hi-Mod 1     $6,390.00 $6,390    $6,390
##  7 Mountain   Over Mountain      Carbon 2     $6,020    $6,070    $5,970
##  8 Mountain   Cross Country Race Carbon 2     $5,595.00 $6,390    $4,800
##  9 Road       Elite Road         Supersix Evo $5,491.00 $12,790   $1,840
## 10 Mountain   Cross Country Race Carbon 3     $5,330.00 $5,330    $5,330
## # ... with 26 more rows
```