101_wk5_functional_program

Seung Hyun Sung

11/10/2021

DS4B 101-R: R FOR BUSINESS ANALYSIS

FUNCTIONAL PROGRAMMING

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1

## Warning: package 'tibble' was built under R version 4.1.1

## Warning: package 'tidyr' was built under R version 4.1.1

## Warning: package 'forcats' was built under R version 4.1.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)
```

```
## Warning: package 'lubridate' was built under R version 4.1.1

##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(tidyquant)
```

```
## Loading required package: PerformanceAnalytics
```

```
## Loading required package: xts
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
##
## Attaching package: 'xts'
```

```
## The following objects are masked from 'package:dplyr':
##
##     first, last
```

```
##
## Attaching package: 'PerformanceAnalytics'
```

```
## The following object is masked from 'package:graphics':
##
##     legend
```

```
## Loading required package: quantmod
```

```
## Loading required package: TTR
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
## == Need to Learn tidyquant? ============================================
## Business Science offers a 1-hour course - Learning Lab #9: Performance Analysis & Portfolio Optimiza
## </> Learn more at: https://university.business-science.io/p/learning-labs-pro </>
```

```
library(ggrepel) # ggrepel needed for text and label repel in plots
```

```
## Warning: package 'ggrepel' was built under R version 4.1.1
```

```
library(fs)
library(kableExtra)
```

```
##
## Attaching package: 'kableExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     group_rows
```

```
bike_orderlines_tbl <- read_rds("~/Desktop/University_business_science/DS4B_101/00_data/bike_sales/data_

glimpse(bike_orderlines_tbl)
```

```
## Rows: 15,644
## Columns: 13
## $ order_date     <dttm> 2011-01-07, 2011-01-07, 2011-01-10, 2011-01-10, 2011-0~
## $ order_id       <dbl> 1, 1, 2, 2, 3, 3, 3, 3, 3, 4, 5, 5, 5, 5, 6, 6, 6, 6, 7~
## $ order_line     <dbl> 1, 2, 1, 2, 1, 2, 3, 4, 5, 1, 1, 2, 3, 4, 1, 2, 3, 4, 1~
## $ quantity       <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1~
## $ price          <dbl> 6070, 5970, 2770, 5970, 10660, 3200, 12790, 5330, 1570,~
## $ total_price    <dbl> 6070, 5970, 2770, 5970, 10660, 3200, 12790, 5330, 1570,~
## $ model          <chr> "Jekyll Carbon 2", "Trigger Carbon 2", "Beast of the Ea~
## $ category_1     <chr> "Mountain", "Mountain", "Mountain", "Mountain", "Road",~
## $ category_2     <chr> "Over Mountain", "Over Mountain", "Trail", "Over Mounta~
## $ frame_material <chr> "Carbon", "Carbon", "Aluminum", "Carbon", "Carbon", "Ca~
## $ bikeshop_name  <chr> "Ithaca Mountain Climbers", "Ithaca Mountain Climbers",~
## $ city           <chr> "Ithaca", "Ithaca", "Kansas City", "Kansas City", "Loui~
## $ state          <chr> "NY", "NY", "KS", "KS", "KY", "KY", "KY", "KY", "KY", "~
```

1.0 ANATOMY OF A FUNCTION —-

1.1 Examining the mean() function —-

```
x <- c(0:10, 50, NA_real_)
x
```

```
##  [1]  0  1  2  3  4  5  6  7  8  9 10 50 NA
```

1.2 Customizing a mean function —-

```
# Name                      # Arguments
mean_remove_na <- function(x, na.rm = TRUE, ...) {

    # Body
    avg <- mean(x, na.rm = na.rm, ...)

    # Return
    return(avg)
```

```
}

mean_remove_na(x)
```

```
## [1] 8.75
```

```
mean_remove_na(x, na.rm = FALSE)
```

```
## [1] NA
```

```
mean_remove_na(x, trim = 0.1)
```

```
## [1] 5.5
```

2.0 THE TWO STYLES OF FUNCTIONS: VECTOR FUNCTIONS & DATA FUNCTIONS —-

Calculating a 3 month rolling average for category_1 & category_2

with dates aligned at last day of the month

```
rolling_avg_3_tbl <- bike_orderlines_tbl %>%
    select(order_date, category_1, category_2, total_price) %>%
    mutate(order_date = ymd(order_date)) %>%
    mutate(month_end = ceiling_date(order_date, unit = "month") - period(1, unit = "day")) %>%
    group_by(category_1, category_2, month_end) %>%
    summarise(
        total_price = sum(total_price)
    ) %>%
    mutate(rolling_avg_3 = rollmean(total_price, k = 3, na.pad = TRUE, align = "right")) %>%
    ungroup() %>%
    mutate(category_2 = as_factor(category_2) %>% fct_reorder2(month_end, total_price))
```

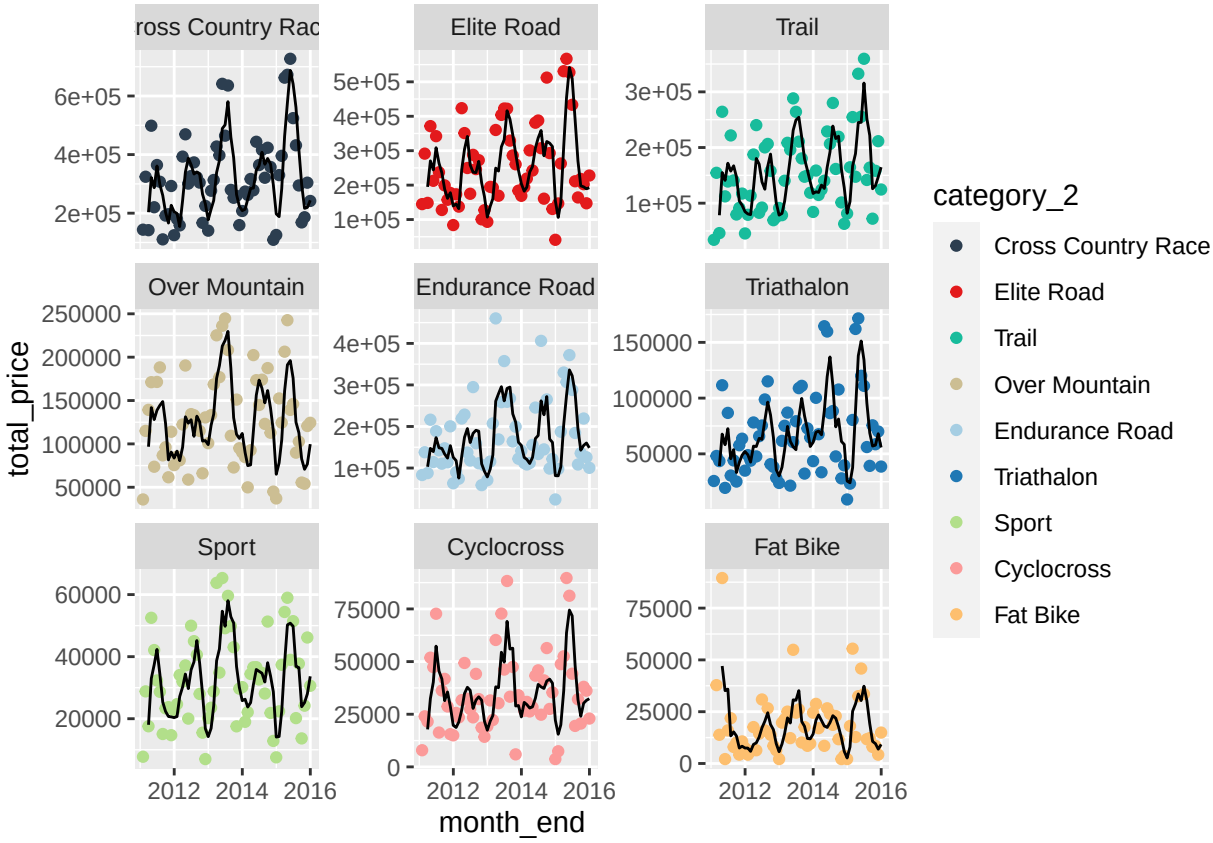
```
## `summarise()` has grouped output by 'category_1', 'category_2'. You can override using the `.groups`
```

```
rolling_avg_3_tbl %>% head(10) %>% kbl() %>% kableExtra::kable_classic()
```

```
rolling_avg_3_tbl %>%
    ggplot(aes(x = month_end, y = total_price)) +
    geom_point(aes(colour = category_2)) +
    geom_line(aes(y = rolling_avg_3), size =0.5) +
    facet_wrap(~category_2, scales = "free_y") +
    scale_color_tq()
```

```
## Warning: Removed 2 row(s) containing missing values (geom_path).
```

category_1	category_2	month_end	total_price	rolling_avg_3
Mountain	Cross Country Race	2011-01-31	143660	NA
Mountain	Cross Country Race	2011-02-28	324400	NA
Mountain	Cross Country Race	2011-03-31	142000	203353.3
Mountain	Cross Country Race	2011-04-30	498580	321660.0
Mountain	Cross Country Race	2011-05-31	220310	286963.3
Mountain	Cross Country Race	2011-06-30	364420	361103.3
Mountain	Cross Country Race	2011-07-31	307300	297343.3
Mountain	Cross Country Race	2011-08-31	110600	260773.3
Mountain	Cross Country Race	2011-09-30	191870	203256.7
Mountain	Cross Country Race	2011-10-31	196440	166303.3



Vectorized & data frame function

Pro Tip: Vectorized functions can be used within mutate() and summarise() functions

Rule of thumb:

- vectorized function commonly starts with x
- data function commonly starts with data
- Any function that get piped into dplyr operation = data frame operation
  - group_by & mutate is data frame operation

  - group_by(.data, ..., .add = FALSE, .drop = group_by_drop_default(.data))
  - Tidy eval operation need to be learned

- Any function that get piped into mutate operation = vectorized operation
  - rollmean is vectorized operation
  - more flexible and easier to make

Controlling Flow:

- Great for checking user input to functions
- Great for Descriptive Messages, Warnings, & Errors

2.1 Vector Functions —-

```
?ymd
?ceiling_date
?sum
?rollmean
```

2.2 Data Functions —-

```
?select
?mutate
?group_by
?ggplot
```

3.0 CONTROLLING FLOW: IF STATEMENTS, MESSAGES, WARNINGS, STOP —-

- Warning: Allows function to continue
- Errors(Stop): Does not allow function to continue

```
class_detect <- function(x){
    if(is.numeric(x)){
        message("Value is numeric")
        print(x)
    } else if(is.character(x)){
        warning("In class detect(): Value is character! Should be numeric, but can be accepted", call. =
        print(x)
    } else if(is.logical(x)){
        stop("In class_detect(): Vlaue is logical!!! Should be numeric. Definitely cannot be accepted",
        print(x)
    } else {
        message("Unknow Class")
```

```
        print(x)
    }
}

1 %>% class_detect
```

```
## Value is numeric
```

```
## [1] 1
```

```
"a" %>% class_detect
```

```
## Warning: In class detect(): Value is character! Should be numeric, but can be
## accepted
```

```
## [1] "a"
```

```
formula(y ~ x) %>% class_detect()
```

```
## Unknow Class
```

```
## y ~ x
```

4.0 VECTORIZED REMOVE OUTLIERS FUNCTION —-

- Box Plot Diagram to Identify Outliers

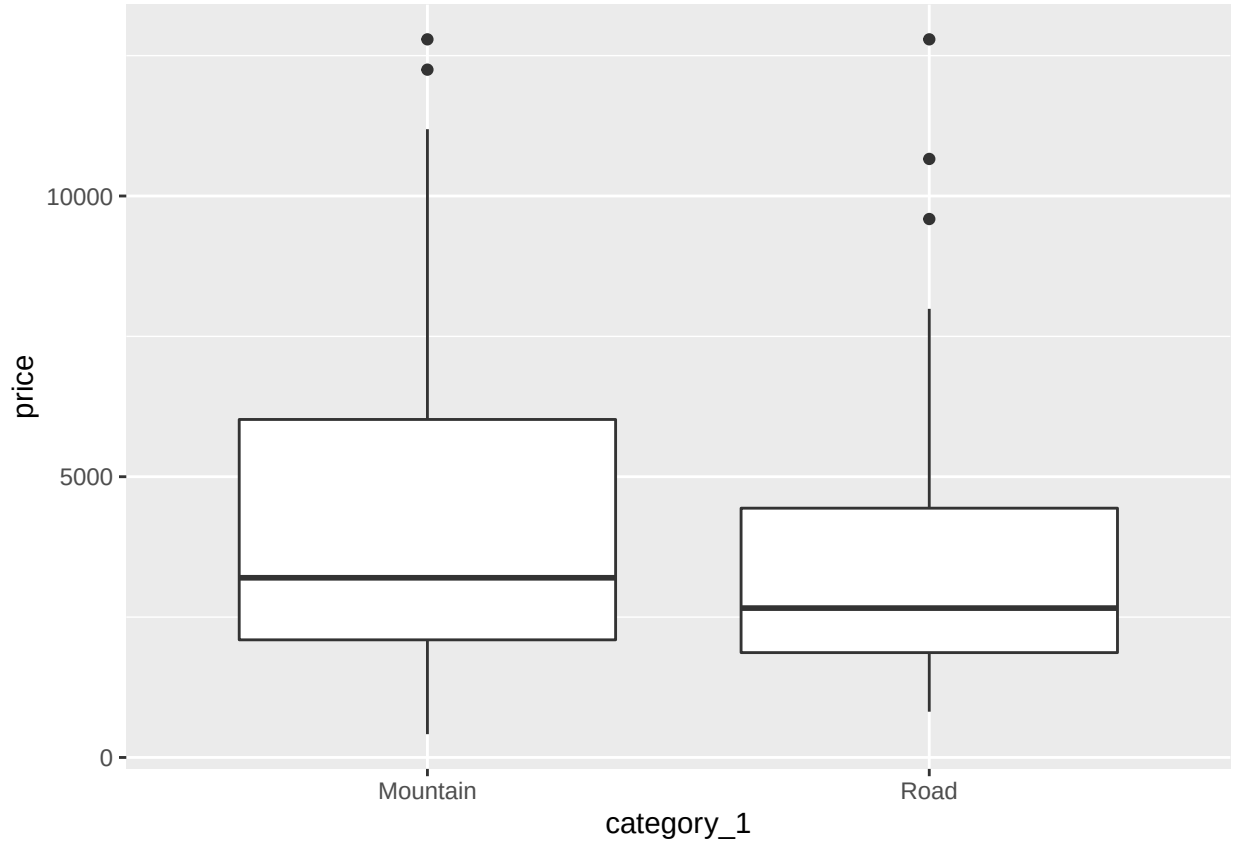
- Goal: Use box plot approach to identify outliers

Make bikes_tbl

```
bikes_tbl <- bike_orderlines_tbl %>%
    distinct(model, category_1, price)
```

Visualize Box Plot

```
bikes_tbl %>%
    ggplot(aes(x = category_1, y = price)) +
    geom_boxplot()
```



Create remove_outliers()

```
# NA_real_: class numeric NA
x <- c(0:10, 50, NA_real_)
x
```

```
##  [1]  0  1  2  3  4  5  6  7  8  9 10 50 NA
```

```
detect_outliers <- function(x){
    if(missing(x)) stop("The argument x needs a vector.")
    if(!is.numeric(x)) stop("The argument x must be numeric.")

    data_tbl <- tibble(data = x)
    limits_tbl <- data_tbl %>%
        summarise(
            quantile_lo = quantile(data, probs = 0.25, na.rm = TRUE),
            quantile_hi = quantile(data, probs = 0.75, na.rm = TRUE),
            iqr = IQR(data, na.rm = TRUE),
            limit_lo = quantile_lo - 1.5*iqr,
            limit_hi = quantile_hi + 1.5*iqr
        )
    output_tbl <- data_tbl %>%
        mutate(outlier = case_when(
```

```
            data < limits_tbl$limit_lo ~ TRUE,
            data > limits_tbl$limit_hi ~ TRUE,
            TRUE ~ FALSE
        ))
    return(output_tbl$outlier)
}

detect_outliers(x)
```

```
##  [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
## [13] FALSE
```

```
tibble(x = x) %>%
    mutate(outlier = detect_outliers(x))
```

```
## # A tibble: 13 x 2
##        x outlier
##    <dbl> <lgl>
##  1     0 FALSE
##  2     1 FALSE
##  3     2 FALSE
##  4     3 FALSE
##  5     4 FALSE
##  6     5 FALSE
##  7     6 FALSE
##  8     7 FALSE
##  9     8 FALSE
## 10     9 FALSE
## 11    10 FALSE
## 12    50 TRUE
## 13    NA FALSE
```

Apply remove_outliers() to bikes_tbl

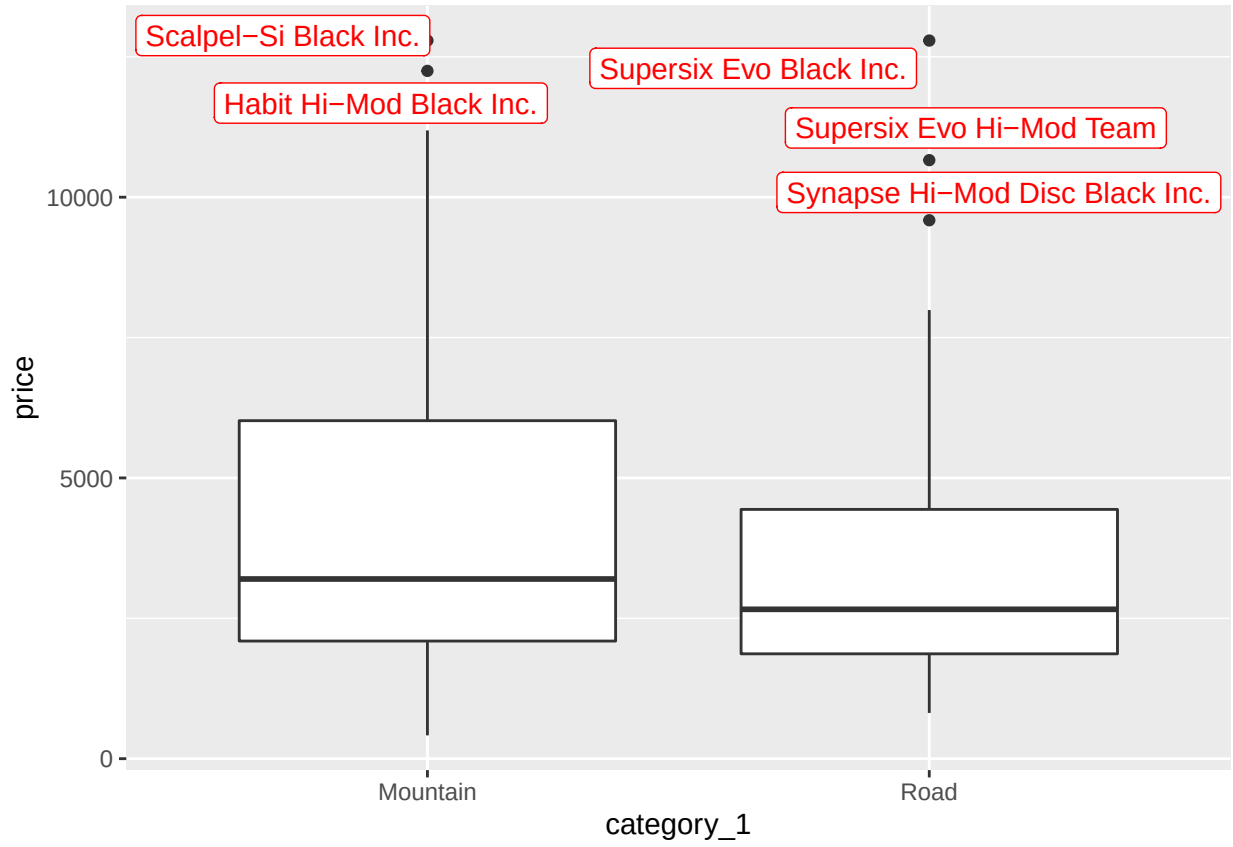
```
bike_outliers_tbl <- bikes_tbl %>%
    group_by(category_1) %>%
    mutate(outlier = detect_outliers(price)) %>%
    ungroup()

bike_outliers_tbl %>% head(10) %>% kbl() %>% kable_classic()
```

Visualize with remove_outlers()

geom_text_repel adds text directly to the plot. geom_label_repel draws a rectangle underneath the text, making it easier to read. The text labels repel away from each other and away from the data points.

price	model	category_1	outlier
6070	Jekyll Carbon 2	Mountain	FALSE
5970	Trigger Carbon 2	Mountain	FALSE
2770	Beast of the East 1	Mountain	FALSE
10660	Supersix Evo Hi-Mod Team	Road	TRUE
3200	Jekyll Carbon 4	Mountain	FALSE
12790	Supersix Evo Black Inc.	Road	TRUE
5330	Supersix Evo Hi-Mod Dura Ace 2	Road	FALSE
1570	Synapse Disc 105	Road	FALSE
4800	Synapse Carbon Disc Ultegra D12	Road	FALSE
480	Catalyst 3	Mountain	FALSE

```
bike_outliers_tbl %>%
    ggplot(aes(category_1, price)) +
    geom_boxplot() +
    ggrepel::geom_label_repel(aes(label = model),
                              color = "red",
                              data = . %>%
                                  filter(outlier))
```



5.0 DATA FUNCTION: FEATURE ENGINEERING —-

- Goal: Want to simplify the text feature engineering steps to convert model name to features

Pipeline Comes From 02_data_wrangling/04_text.R

```
bikes_tbl %>%

    select(model) %>%

    # Fix typo
    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
    )) %>%

    # separate using spaces
    separate(col     = model,
             into    = str_c("model_", 1:7),
             sep     = " ",
             remove  = FALSE,
             fill    = "right") %>%

    # creating a "base" feature
    mutate(model_base = case_when(

        # 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 Beast of the East
        str_detect(str_to_lower(model_1), "beast") ~ str_c(model_1, model_2, model_3, model_4, sep = "

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

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

        # catch all
        TRUE ~ model_1)
    ) %>%

    # Get "tier" feature
    mutate(model_tier = model %>% str_replace(model_base, replacement = "") %>% str_trim()) %>%

    # Remove unnecessary columns
    select(-matches("[0-9]")) %>%

    # Create Flags
    mutate(
        black     = model_tier %>% str_to_lower() %>% str_detect("black") %>% as.numeric(),
        hi_mod    = model_tier %>% str_to_lower() %>% str_detect("hi-mod") %>% as.numeric(),
        team      = model_tier %>% str_to_lower() %>% str_detect("team") %>% as.numeric(),
```

```
        red       = model_tier %>% str_to_lower() %>% str_detect("red") %>% as.numeric(),
        ultegra   = model_tier %>% str_to_lower() %>% str_detect("ultegra") %>% as.numeric(),
        dura_ace  = model_tier %>% str_to_lower() %>% str_detect("dura ace") %>% as.numeric(),
        disc      = model_tier %>% str_to_lower() %>% str_detect("disc") %>% as.numeric()
    )
```

```
## # A tibble: 97 x 10
##    model   model_base model_tier black hi_mod  team   red ultegra dura_ace  disc
##    <chr>   <chr>      <chr>      <dbl>  <dbl> <dbl> <dbl>   <dbl>    <dbl> <dbl>
##  1 Jekyll~ Jekyll     Carbon 2       0      0     0     0       0        0     0
##  2 Trigge~ Trigger    Carbon 2       0      0     0     0       0        0     0
##  3 Beast ~ Beast of ~ 1              0      0     0     0       0        0     0
##  4 Supers~ Supersix ~ Hi-Mod Te~     0      1     1     0       0        0     0
##  5 Jekyll~ Jekyll     Carbon 4       0      0     0     0       0        0     0
##  6 Supers~ Supersix ~ Black Inc.     1      0     0     0       0        0     0
##  7 Supers~ Supersix ~ Hi-Mod Du~     0      1     0     0       0        1     0
##  8 Synaps~ Synapse    Disc 105       0      0     0     0       0        0     1
##  9 Synaps~ Synapse    Carbon Di~     0      0     0     0       1        0     1
## 10 Cataly~ Catalyst   3              0      0     0     0       0        0     0
## # ... with 87 more rows
```

```
data <- bikes_tbl

separate_bike_model <- function(data, keep_model_column = TRUE, append = TRUE){

    # Append argument
    if(!append){
        data <- data %>% select(model)
    }
    # pipline
    output_tbl <- data %>%

    #select(model) %>%

    # Fix typo
    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
    )) %>%

    # separate using spaces
    separate(col     = model,
             into    = str_c("model_", 1:7),
             sep     = " ",
             remove  = FALSE,
             fill    = "right") %>%

    # creating a "base" feature
    mutate(model_base = case_when(
        # Fix Supersix Evo
        str_detect(str_to_lower(model_1), "supersix") ~ str_c(model_1, model_2, sep = " "),
```

price	model	category_1	model_base	model_tier	black	hi_mo
6070	Jekyll Carbon 2	Mountain	Jekyll	Carbon 2	0	
5970	Trigger Carbon 2	Mountain	Trigger	Carbon 2	0	
2770	Beast of the East 1	Mountain	Beast of the East	1	0	
10660	Supersix Evo Hi-Mod Team	Road	Supersix Evo	Hi-Mod Team	0	
3200	Jekyll Carbon 4	Mountain	Jekyll	Carbon 4	0	
12790	Supersix Evo Black Inc.	Road	Supersix Evo	Black Inc.	1	
5330	Supersix Evo Hi-Mod Dura Ace 2	Road	Supersix Evo	Hi-Mod Dura Ace 2	0	
1570	Synapse Disc 105	Road	Synapse	Disc 105	0	
4800	Synapse Carbon Disc Ultegra D12	Road	Synapse	Carbon Disc Ultegra D12	0	
480	Catalyst 3	Mountain	Catalyst	3	0	

```
        # Fix Fat CAAD bikes
        str_detect(str_to_lower(model_1), "fat") ~ str_c(model_1, model_2, sep = " "),
        # 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 Bad Habit
        str_detect(str_to_lower(model_1), "bad") ~ str_c(model_1, model_2, sep = " "),
        # Fix Scalpel 29
        str_detect(str_to_lower(model_2), "29") ~ str_c(model_1, model_2, sep = " "),
        # catch all
        TRUE ~ model_1)
    ) %>%
    # Get "tier" feature
    mutate(model_tier = model %>% str_replace(model_base, replacement = "") %>% str_trim()) %>%
    # Remove unnecessary columns
    select(-matches("model_[0-9]")) %>%
    # Create Flags
    mutate(
        black     = model_tier %>% str_to_lower() %>% str_detect("black") %>% as.numeric(),
        hi_mod    = model_tier %>% str_to_lower() %>% str_detect("hi-mod") %>% as.numeric(),
        team      = model_tier %>% str_to_lower() %>% str_detect("team") %>% as.numeric(),
        red       = model_tier %>% str_to_lower() %>% str_detect("red") %>% as.numeric(),
        ultegra   = model_tier %>% str_to_lower() %>% str_detect("ultegra") %>% as.numeric(),
        dura_ace  = model_tier %>% str_to_lower() %>% str_detect("dura ace") %>% as.numeric(),
        disc      = model_tier %>% str_to_lower() %>% str_detect("disc") %>% as.numeric()
    )

    if(!keep_model_column) output_tbl <- output_tbl %>% select(-model)

    return(output_tbl)
}

bikes_tbl %>% separate_bike_model() %>% head(10) %>% kbl() %>% kable_classic()
```

```
bikes_tbl %>% separate_bike_model(keep_model_column = FALSE)  %>% head(10) %>% kbl() %>% kable_classic()
```

```
bikes_tbl %>% separate_bike_model(keep_model_column = FALSE, append = FALSE)  %>% head(10) %>% kbl() %>%
```

price	category_1	model_base	model_tier	black	hi_mod	team	red	ultegra	dura_ace
6070	Mountain	Jekyll	Carbon 2	0	0	0	0	0	0
5970	Mountain	Trigger	Carbon 2	0	0	0	0	0	0
2770	Mountain	Beast of the East	1	0	0	0	0	0	0
10660	Road	Supersix Evo	Hi-Mod Team	0	1	1	0	0	0
3200	Mountain	Jekyll	Carbon 4	0	0	0	0	0	0
12790	Road	Supersix Evo	Black Inc.	1	0	0	0	0	0
5330	Road	Supersix Evo	Hi-Mod Dura Ace 2	0	1	0	0	0	1
1570	Road	Synapse	Disc 105	0	0	0	0	0	0
4800	Road	Synapse	Carbon Disc Ultegra D12	0	0	0	0	1	0
480	Mountain	Catalyst	3	0	0	0	0	0	0

model_base	model_tier	black	hi_mod	team	red	ultegra	dura_ace	disc
Jekyll	Carbon 2	0	0	0	0	0	0	0
Trigger	Carbon 2	0	0	0	0	0	0	0
Beast of the East	1	0	0	0	0	0	0	0
Supersix Evo	Hi-Mod Team	0	1	1	0	0	0	0
Jekyll	Carbon 4	0	0	0	0	0	0	0
Supersix Evo	Black Inc.	1	0	0	0	0	0	0
Supersix Evo	Hi-Mod Dura Ace 2	0	1	0	0	0	1	0
Synapse	Disc 105	0	0	0	0	0	0	1
Synapse	Carbon Disc Ultegra D12	0	0	0	0	1	0	1
Catalyst	3	0	0	0	0	0	0	0

6.0 SAVING AND SOURCING FUNCTIONS —-

6.1 Create folder and file —-

```
fs::dir_create("00")

path <- "01_Scripts/separate_bikes_and_outlier_detection.R"
fs::file_create(path)
```

6.2 Build and add header —-

```
file_header_text <- str_glue(
"
# SEPARATE BIKE MODELS AND DETECT OUTLIERS ----

# separate_bikes_models(): A tidy function to separate the model column into engineered features

# detect_outliers(): A vectorized function that detects outliers using TRUE/FALSE output

# Libraries ----
library(tidyverse)
```

```
"
)

write_lines(file_header_text, path)
```

6.3 Add functions with dump() —-

```
c("separate_bike_model", "detect_outliers") %>%
    dump(file = "01_Scripts/separate_bikes_and_outlier_detection.R",
         append = TRUE)
```

6.4 Source function —-

```
rm("separate_bike_model")

source("01_Scripts/separate_bikes_and_outlier_detection.R")
```