Data Wrangling — bikes example

AUTHOR
Anungoo-0914

PUBLISHED
December 19, 2025

1. Install packages

(Only run the install lines once on your machine. In this document they are shown with `eval = FALSE` so they won't run automatically when knitting.)

```
install.packages("readxl")
install.packages("tidyverse") # Only run once
install.packages("tidyquant")
```

2. Libraries

```
library(tidyverse)
library(tidyquant)
library(readxl)
library(writexl)
library(lubridate)
```

3. Data import

Adjust paths if necessary. The example uses the `./FDB_2025F/` directory.

```
bikes_tbl <- read_excel("./bikes.xlsx") # fast key: alt+-
bikeshops_tbl <- read_excel("./bikeshops.xlsx")
orderlines_tbl <- read_excel("./orderlines.xlsx")

# Examine data
bikes_tbl
```

```
# A tibble: 97 × 4
   bike.id model                          description                price
     <dbl> <chr>                          <chr>                      <dbl>
 1       1 Supersix Evo Black Inc.        Road - Elite Road - Carbon 12790
 2       2 Supersix Evo Hi-Mod Team       Road - Elite Road - Carbon 10660
 3       3 Supersix Evo Hi-Mod Dura Ace 1 Road - Elite Road - Carbon  7990
 4       4 Supersix Evo Hi-Mod Dura Ace 2 Road - Elite Road - Carbon  5330
 5       5 Supersix Evo Hi-Mod Utegra     Road - Elite Road - Carbon  4260
 6       6 Supersix Evo Red               Road - Elite Road - Carbon  3940
 7       7 Supersix Evo Ultegra 3         Road - Elite Road - Carbon  3200
 8       8 Supersix Evo Ultegra 4         Road - Elite Road - Carbon  2660
```

```
 9       9 Supersix Evo 105              Road - Elite Road - Carbon  2240
10      10 Supersix Evo Tiagra           Road - Elite Road - Carbon  1840
# ℹ 87 more rows
```

```
head(bikes_tbl)
```

```
# A tibble: 6 × 4
  bike.id model                          description               price
    <dbl> <chr>                          <chr>                     <dbl>
1       1 Supersix Evo Black Inc.        Road - Elite Road - Carbon 12790
2       2 Supersix Evo Hi-Mod Team       Road - Elite Road - Carbon 10660
3       3 Supersix Evo Hi-Mod Dura Ace 1 Road - Elite Road - Carbon  7990
4       4 Supersix Evo Hi-Mod Dura Ace 2 Road - Elite Road - Carbon  5330
5       5 Supersix Evo Hi-Mod Utegra     Road - Elite Road - Carbon  4260
6       6 Supersix Evo Red               Road - Elite Road - Carbon  3940
```

```
# Import csv file:
bike_orderlines_tbl <- read_csv("./bike_orderlines.csv")
```

4. Joining data

```
# Joining data:
orderlines_bikes_tbl <- left_join(orderlines_tbl, bikes_tbl, by = c("product.id" = "bike.id"))

bike_orderlines_bikeshops_joined <- left_join(orderlines_bikes_tbl, bikeshops_tbl,
                                              by = c('customer.id' = 'bikeshop.id'))

# Equivalent pipe chain:
bike_orderlines_bikeshops_joined <- left_join(orderlines_tbl, bikes_tbl, by = c("product.id" = "b
                                    left_join(bikeshops_tbl, by = c("customer.id" = "bikeshop.id"
```

5. Wrangling data

Decompose `description`, separate `location`, compute `total.price`, rename and clean names, and save as RDS.

```
bike_orderlines_wrangled_tbl <- bike_orderlines_bikeshops_joined %>%
  separate(description,
           into = c('category.1', 'category.2', 'frame.material'),
           sep  = ' - ') %>%
  separate(location,
           into = c('city', 'state'),
           sep  = ', ',
           remove = FALSE) %>%
  # create calculated columns
```

```
  mutate(total.price = price * quantity) %>%
  # Reorganize columns
  select(-...1, -location) %>%
  # Reorder columns
  select(contains('date'), contains('id'),
         contains('order'),
         quantity, price, total.price,
         everything()) %>%
  # Rename columns
  rename(order_date = order.date) %>%
  set_names(names(.) %>% str_replace_all("\\.", "_"))

# save the file as RDS
saveRDS(bike_orderlines_wrangled_tbl, './bike_orderlines.rds')
```

6. dplyr / tidyr examples

Examples showing pull(), select_if(), arrange(), filter(), distinct(), mutate(), ntile(), case_when(), summarise(), group_by().

```
# pull() vs select()
bike_orderlines_wrangled_tbl %>%
  # select(total_price)
  pull(total_price) %>%
  mean()
```

```
[1] 4540.548
```

```
# select_if
bike_orderlines_wrangled_tbl %>%
  # select_if(is.character)
  select_if(is.numeric)
```

```
# A tibble: 15,644 × 7
   order_id customer_id product_id order_line quantity price total_price
      <dbl>       <dbl>      <dbl>      <dbl>    <dbl> <dbl>       <dbl>
 1        1           2         48          1        1  6070        6070
 2        1           2         52          2        1  5970        5970
 3        2          10         76          1        1  2770        2770
 4        2          10         52          2        1  5970        5970
 5        3           6          2          1        1 10660       10660
 6        3           6         50          2        1  3200        3200
 7        3           6          1          3        1 12790       12790
 8        3           6          4          4        1  5330        5330
 9        3           6         34          5        1  1570        1570
10        4          22         26          1        1  4800        4800
# ℹ 15,634 more rows
```

```
# arrange() and desc()
bikes_tbl %>%
  select(model, price) %>%
  arrange(desc(price))
```

```
# A tibble: 97 × 2
   model                          price
   <chr>                          <dbl>
 1 Supersix Evo Black Inc.        12790
 2 Scalpel-Si Black Inc.          12790
 3 Habit Hi-Mod Black Inc.        12250
 4 F-Si Black Inc.                11190
 5 Supersix Evo Hi-Mod Team       10660
 6 Synapse Hi-Mod Disc Black Inc.  9590
 7 Scalpel-Si Race                 9060
 8 F-Si Hi-Mod Team                9060
 9 Trigger Carbon 1                8200
10 Supersix Evo Hi-Mod Dura Ace 1  7990
# ℹ 87 more rows
```

```
# filter()
bikes_tbl %>%
  select(model, price) %>%
  filter(price > mean(price))
```

```
# A tibble: 35 × 2
   model                          price
   <chr>                          <dbl>
 1 Supersix Evo Black Inc.        12790
 2 Supersix Evo Hi-Mod Team       10660
 3 Supersix Evo Hi-Mod Dura Ace 1  7990
 4 Supersix Evo Hi-Mod Dura Ace 2  5330
 5 Supersix Evo Hi-Mod Utegra      4260
 6 CAAD12 Black Inc                5860
 7 CAAD12 Disc Dura Ace            4260
 8 Synapse Hi-Mod Disc Black Inc.  9590
 9 Synapse Hi-Mod Disc Red         7460
10 Synapse Hi-Mod Dura Ace         5860
# ℹ 25 more rows
```

```
bikes_tbl %>%
  select(model, price) %>%
  filter((price > 5000) & (price < 10000)) %>%
  arrange(desc(price))
```

```
# A tibble: 22 × 2
   model                          price
   <chr>                          <dbl>
 1 Synapse Hi-Mod Disc Black Inc.  9590
```

```
 2 Scalpel-Si Race                 9060
 3 F-Si Hi-Mod Team                9060
 4 Trigger Carbon 1                8200
 5 Supersix Evo Hi-Mod Dura Ace 1  7990
 6 Jekyll Carbon 1                 7990
 7 Synapse Hi-Mod Disc Red         7460
 8 Scalpel-Si Hi-Mod 1             7460
 9 Habit Carbon 1                  7460
10 Slice Hi-Mod Black Inc.         7000
# ℹ 12 more rows
```

```
bikes_tbl %>%
  select(model, price) %>%
  filter(price > 6000,
         model %>% str_detect("Supersix"))
```

```
# A tibble: 3 × 2
  model                          price
  <chr>                          <dbl>
1 Supersix Evo Black Inc.        12790
2 Supersix Evo Hi-Mod Team       10660
3 Supersix Evo Hi-Mod Dura Ace 1  7990
```

```
# Filtering one or more conditions using == and %in%
bike_orderlines_wrangled_tbl %>%
  filter(category_2 %in% c("Over Mountain", "Trail", "Endurance Road")) %>%
  View()

# slice()
bikes_tbl %>%
  arrange(desc(price)) %>%
  # slice(1:5)
  slice((nrow(.)-4):nrow(.))
```

```
# A tibble: 5 × 4
  bike.id model      description                price
    <dbl> <chr>      <chr>                      <dbl>
1      93 Trail 5    Mountain - Sport - Aluminum   815
2      94 Catalyst 1 Mountain - Sport - Aluminum   705
3      95 Catalyst 2 Mountain - Sport - Aluminum   585
4      96 Catalyst 3 Mountain - Sport - Aluminum   480
5      97 Catalyst 4 Mountain - Sport - Aluminum   415
```

```
# distinct(): extract unique values from data
bike_orderlines_wrangled_tbl %>%
  distinct(category_1, category_2) %>%
  View()

# mutate(): add new columns
```

```
bike_orderlines_wrangled_tbl %>%
  mutate(total_price_log = log(total_price)) %>%
  mutate(total_price_sqrt = total_price^0.5) %>%
  View()

# Binning with ntile()
bike_orderlines_wrangled_tbl %>%
  mutate(total_price_binned = ntile(total_price, 3)) %>%
  View()

# case_when() example
bike_orderlines_wrangled_tbl %>%
  mutate(total_price_binned = ntile(total_price, 3)) %>%
  mutate(total_price_binned2 = case_when(
    total_price > quantile(total_price, 0.75) ~ "High",
    total_price > quantile(total_price, 0.25) ~ "Medium",
    TRUE ~ "Low"
  )) %>%
  View()
```

7. Grouping & summarizing

```
bike_orderlines_wrangled_tbl %>%
  summarise(revenue = sum(total_price))
```

```
# A tibble: 1 × 1
   revenue
     <dbl>
1 71032330
```

```
bike_orderlines_wrangled_tbl %>%
  group_by(category_1) %>%
  summarise(revenue = sum(total_price)) %>%
  ungroup() %>%
  arrange(desc(revenue))
```

```
# A tibble: 2 × 2
  category_1  revenue
  <chr>         <dbl>
1 Mountain   39154735
2 Road       31877595
```

```
bike_orderlines_wrangled_tbl %>%
  group_by(category_1, category_2, frame_material) %>%
  summarise(revenue = sum(total_price)) %>%
  ungroup() %>%
  arrange(desc(revenue))
```

```
# A tibble: 13 × 4
   category_1 category_2         frame_material  revenue
   <chr>      <chr>              <chr>             <dbl>
 1 Mountain   Cross Country Race Carbon         15906070
 2 Road       Elite Road         Carbon          9696870
 3 Road       Endurance Road     Carbon          8768610
 4 Mountain   Over Mountain      Carbon          7571270
 5 Road       Elite Road         Aluminum        5637795
 6 Mountain   Trail              Carbon          4835850
 7 Mountain   Trail              Aluminum        4537610
 8 Road       Triathalon         Carbon          4053750
 9 Mountain   Cross Country Race Aluminum        3318560
10 Road       Cyclocross         Carbon          2108120
11 Mountain   Sport              Aluminum        1932755
12 Road       Endurance Road     Aluminum        1612450
13 Mountain   Fat Bike           Aluminum        1052620
```

8. Questions & quick answers

```
# Q1: What are the unique categories of products?
bike_orderlines_wrangled_tbl %>% distinct(category_1)
```

```
# A tibble: 2 × 1
  category_1
  <chr>
1 Mountain
2 Road
```

```
bike_orderlines_wrangled_tbl %>% distinct(category_2)
```

```
# A tibble: 9 × 1
  category_2
  <chr>
1 Over Mountain
2 Trail
3 Elite Road
4 Endurance Road
5 Sport
6 Cross Country Race
7 Cyclocross
8 Triathalon
9 Fat Bike
```

```
bike_orderlines_wrangled_tbl %>% distinct(frame_material)
```

```
# A tibble: 2 × 1
  frame_material
  <chr>
```

```
1 Carbon
2 Aluminum
```

```
# Q2: Which product categories have the largest sales? (category_1)
bike_orderlines_wrangled_tbl %>%
  select(category_1, total_price) %>%
  group_by(category_1) %>%
  summarise(sales = sum(total_price)) %>%
  ungroup() %>%
  rename(`Primary Category` = category_1,
         Sales = sales) %>%
  # format dollars
  mutate(Sales1 = Sales %>% scales::dollar())
```

```
# A tibble: 2 × 3
  `Primary Category`    Sales Sales1
  <chr>                 <dbl> <chr>
1 Mountain           39154735 $39,154,735
2 Road               31877595 $31,877,595
```

9. Time series / date handling (lubridate)

```
# Check structure
str(bike_orderlines_wrangled_tbl)
```

```
tibble [15,644 × 15] (S3: tbl_df/tbl/data.frame)
 $ order_date    : POSIXct[1:15644], format: "2011-01-07" "2011-01-07" ...
 $ order_id      : num [1:15644] 1 1 2 2 3 3 3 3 3 4 ...
 $ customer_id   : num [1:15644] 2 2 10 10 6 6 6 6 6 22 ...
 $ product_id    : num [1:15644] 48 52 76 52 2 50 1 4 34 26 ...
 $ order_line    : num [1:15644] 1 2 1 2 1 2 3 4 5 1 ...
 $ quantity      : num [1:15644] 1 1 1 1 1 1 1 1 1 1 ...
 $ price         : num [1:15644] 6070 5970 2770 5970 10660 ...
 $ total_price   : num [1:15644] 6070 5970 2770 5970 10660 ...
 $ model         : chr [1:15644] "Jekyll Carbon 2" "Trigger Carbon 2" "Beast of the East 1"
"Trigger Carbon 2" ...
 $ category_1    : chr [1:15644] "Mountain" "Mountain" "Mountain" "Mountain" ...
 $ category_2    : chr [1:15644] "Over Mountain" "Over Mountain" "Trail" "Over Mountain" ...
 $ frame_material: chr [1:15644] "Carbon" "Carbon" "Aluminum" "Carbon" ...
 $ bikeshop_name : chr [1:15644] "Ithaca Mountain Climbers" "Ithaca Mountain Climbers" "Kansas
City 29ers" "Kansas City 29ers" ...
 $ city          : chr [1:15644] "Ithaca" "Ithaca" "Kansas City" "Kansas City" ...
 $ state         : chr [1:15644] "NY" "NY" "KS" "KS" ...
```

```
bike_sales_y <- bike_orderlines_wrangled_tbl %>%
  select(order_date, total_price) %>%
  # change order_date into ymd format
  mutate(order_date = ymd(order_date)) %>%
```

```
  mutate(year = year(order_date)) %>%
  # group by year
  group_by(year) %>%
  summarise(sales = sum(total_price)) %>%
  ungroup()

# monthly aggregation
bike_sales_m <- bike_orderlines_wrangled_tbl %>%
  select(order_date, total_price) %>%
  mutate(order_date = ymd(order_date)) %>%
  mutate(year_month = floor_date(order_date, unit = "month")) %>%
  group_by(year_month) %>%
  summarise(sales = sum(total_price))
```

10. Measuring change / lag & a helper function

```
# Example mutate with lag (fix variable names from script)
bike_sales_y %>%
  mutate(sales_lag_1 = lag(sales, n = 1)) %>%
  # Replace NA with the first year's sales (if desired)
  mutate(sales_lag_1 = case_when(
    is.na(sales_lag_1) ~ sales,
    TRUE ~ sales_lag_1
  )) %>%
  mutate(diff_1 = sales - sales_lag_1) %>%
  mutate(pct_diff_1 = diff_1 / sales_lag_1) %>%
  mutate(pct_diff_1_chr = scales::percent(pct_diff_1))
```

```
# A tibble: 5 × 6
   year    sales sales_lag_1   diff_1 pct_diff_1 pct_diff_1_chr
  <dbl>    <dbl>       <dbl>    <dbl>      <dbl> <chr>
1  2011 11292885    11292885        0     0      0.0%
2  2012 12163075    11292885   870190     0.0771 7.7%
3  2013 16480775    12163075  4317700     0.355  35.5%
4  2014 13924085    16480775 -2556690    -0.155  -15.5%
5  2015 17171510    13924085  3247425     0.233  23.3%
```

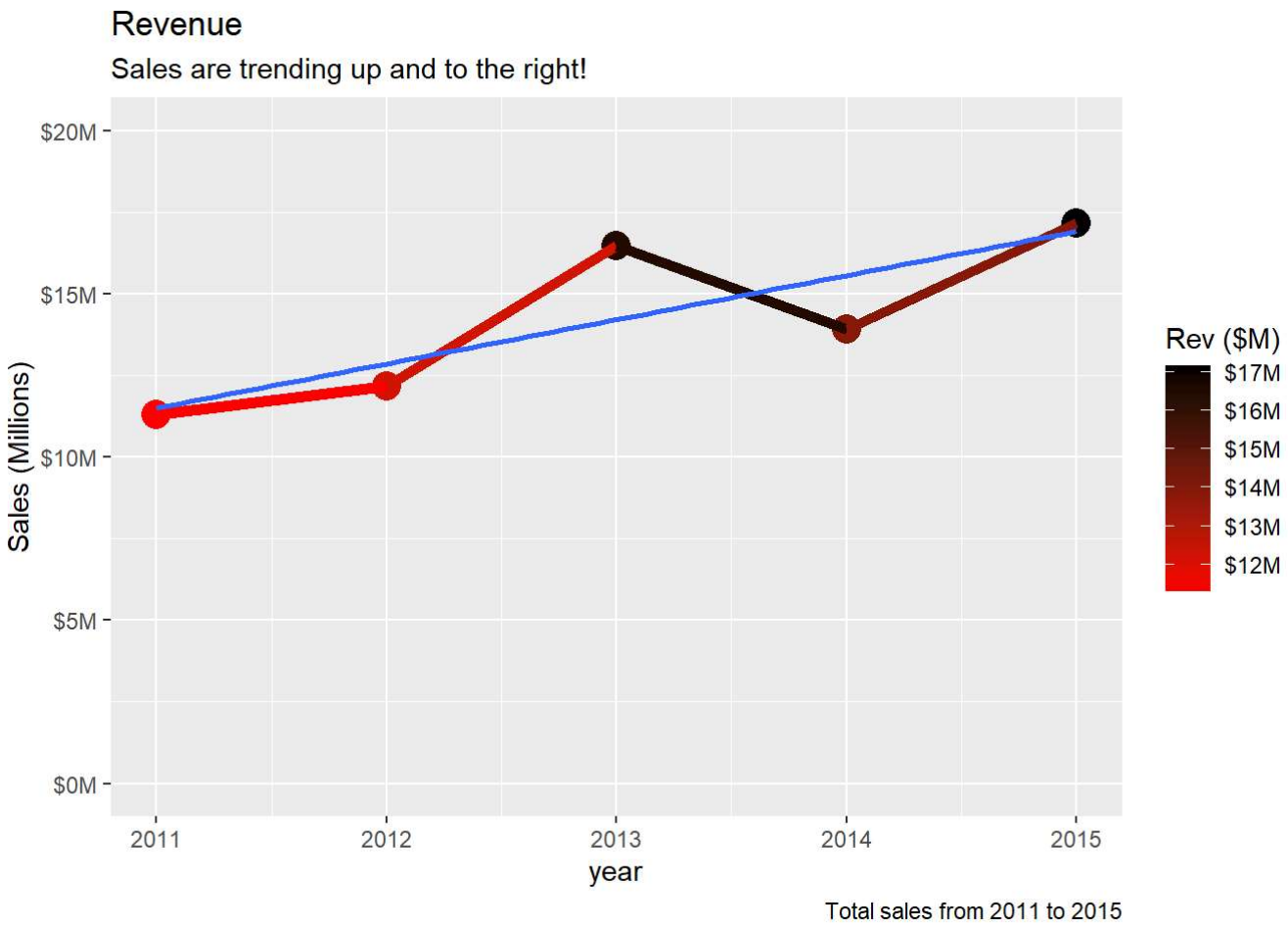
```
# Function to compute percent change
calculate_pct_diff <- function(data){
  data %>%
    mutate(sales_lag_1 = lag(sales, n = 1)) %>%
    mutate(sales_lag_1 = case_when(
      is.na(sales_lag_1) ~ sales,
      TRUE ~ sales_lag_1
    )) %>%
    mutate(diff_1 = sales - sales_lag_1) %>%
    mutate(pct_diff_1 = diff_1 / sales_lag_1) %>%
    mutate(pct_diff_1_chr = scales::percent(pct_diff_1))
```

```
}

calculate_pct_diff(bike_sales_m)
```

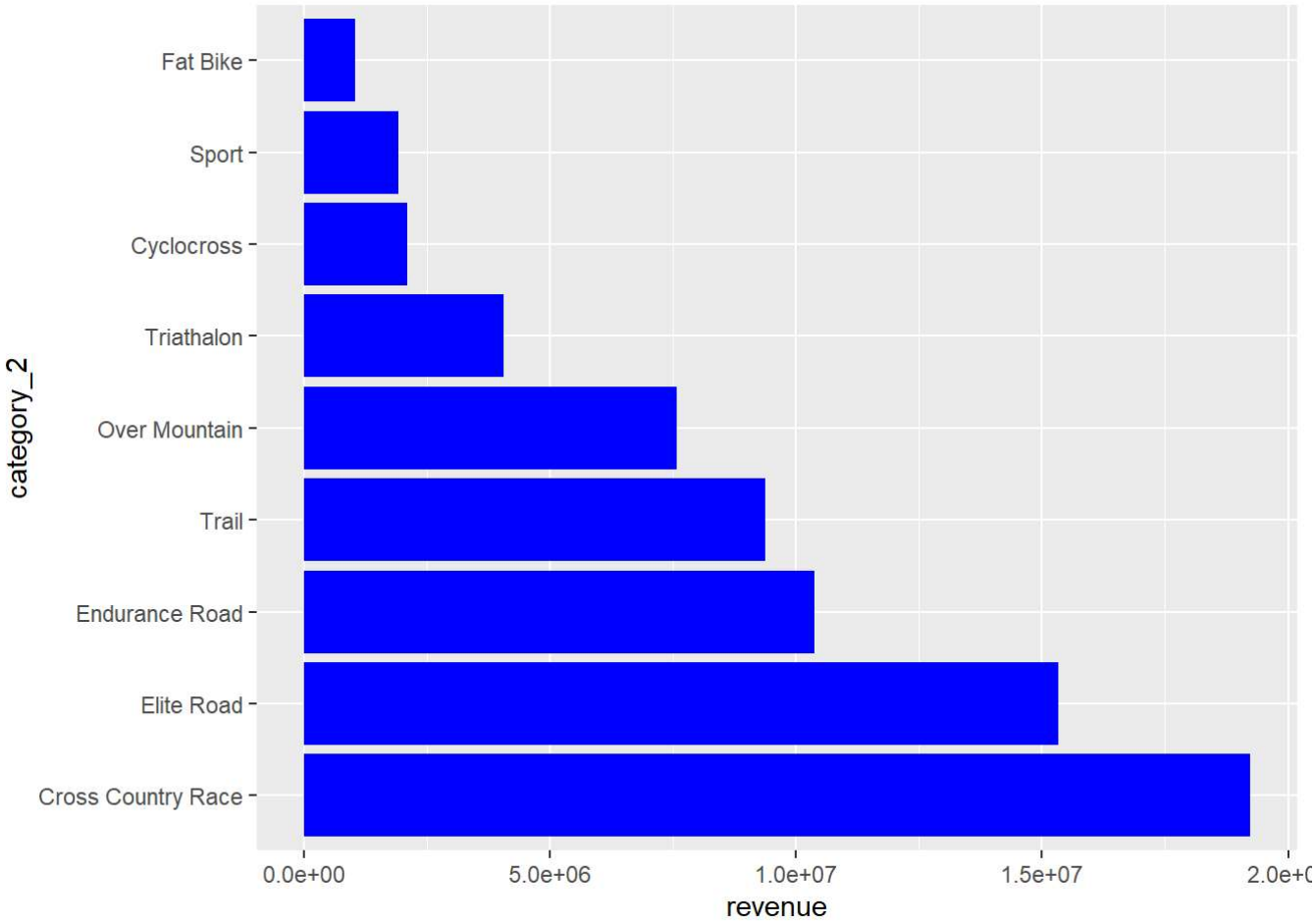
```
# A tibble: 60 × 6
   year_month   sales sales_lag_1  diff_1 pct_diff_1 pct_diff_1_chr
   <date>       <dbl>       <dbl>   <dbl>      <dbl> <chr>
 1 2011-01-01  483015      483015       0     0      0.000%
 2 2011-02-01 1162075      483015  679060     1.41   140.588%
 3 2011-03-01  659975     1162075 -502100    -0.432  -43.207%
 4 2011-04-01 1827140      659975 1167165     1.77   176.850%
 5 2011-05-01  844170     1827140 -982970    -0.538  -53.798%
 6 2011-06-01 1413445      844170  569275     0.674  67.436%
 7 2011-07-01 1194430     1413445 -219015    -0.155  -15.495%
 8 2011-08-01  679790     1194430 -514640    -0.431  -43.087%
 9 2011-09-01  814720      679790  134930     0.198  19.849%
10 2011-10-01  734920      814720  -79800    -0.0979 -9.795%
# ℹ 50 more rows
```

11. Plots (ggplot2 / tidyquant theme)

```
# yearly sales plot
bike_sales_y %>%
  ggplot(aes(x = year, y = sales, color = sales)) +
  geom_point(size = 5) +
  geom_line(linewidth = 2) +
  geom_smooth(method =  "lm", formula = 'y ~ x', se = FALSE) +
  expand_limits(y = c(0, 20e6)) +
  scale_colour_continuous(low = "red", high = "black",
                          labels = scales::dollar_format(scale = 1/1e6, suffix = "M")) +
  scale_y_continuous(labels = scales::dollar_format(scale = 1/1e6, suffix = "M")) +
  labs(
    title    = "Revenue",
    subtitle = "Sales are trending up and to the right!",
    x        = "year",
    y        = "Sales (Millions)",
    color    = "Rev ($M)",
    caption  = "Total sales from 2011 to 2015"
  )
```

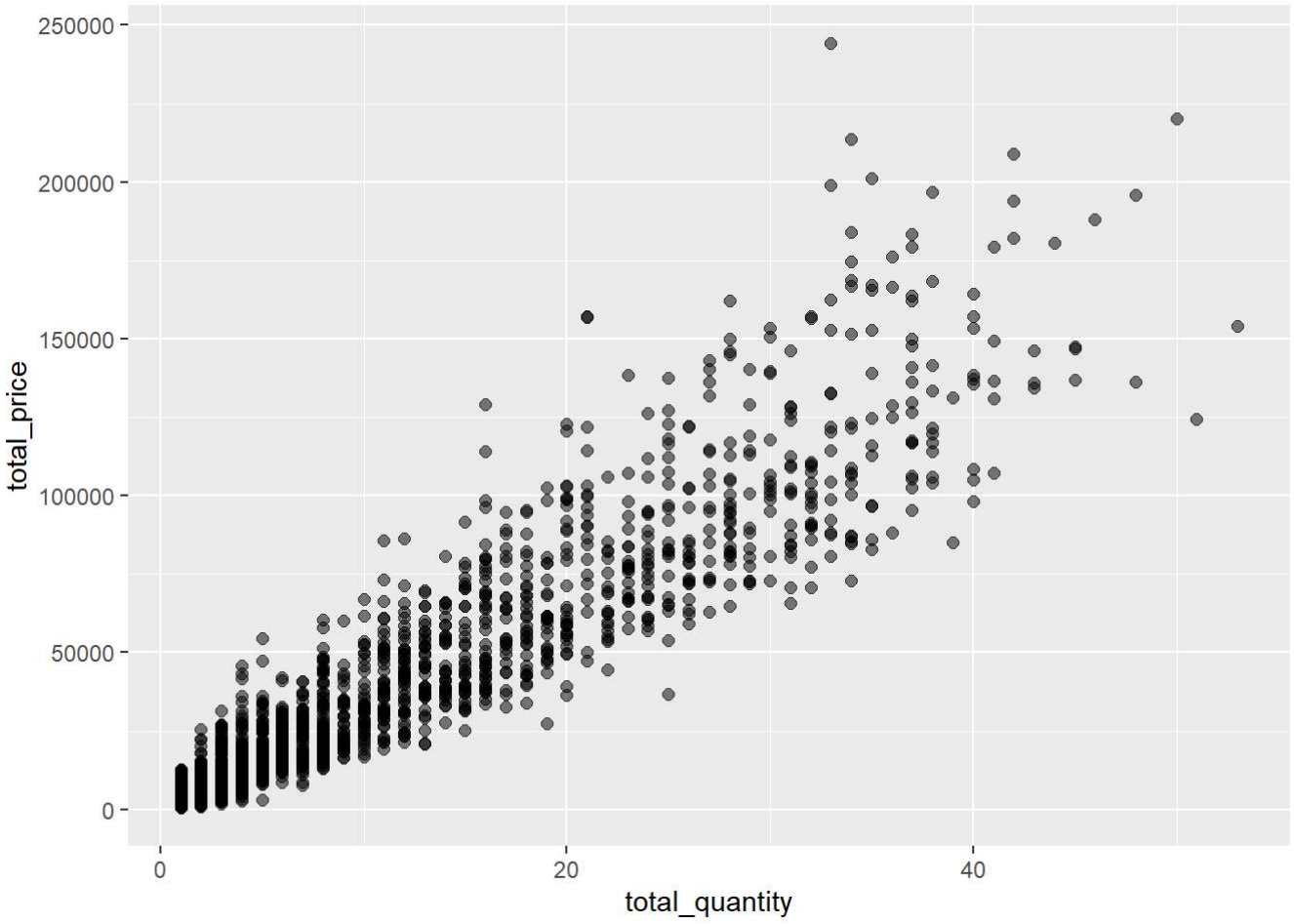



```
# Bar plot: revenue by category_2
revenue_by_category2_tbl <- bike_orderlines_wrangled_tbl %>%
  select(category_2, total_price) %>%
  group_by(category_2) %>%
  summarise(revenue = sum(total_price)) %>%
  ungroup()

revenue_by_category2_tbl %>%
  mutate(category_2 = category_2 %>% as_factor() %>% fct_reorder(desc(revenue))) %>%
  ggplot(aes(category_2, revenue)) +
  geom_col(fill = "blue") +
  coord_flip()
```

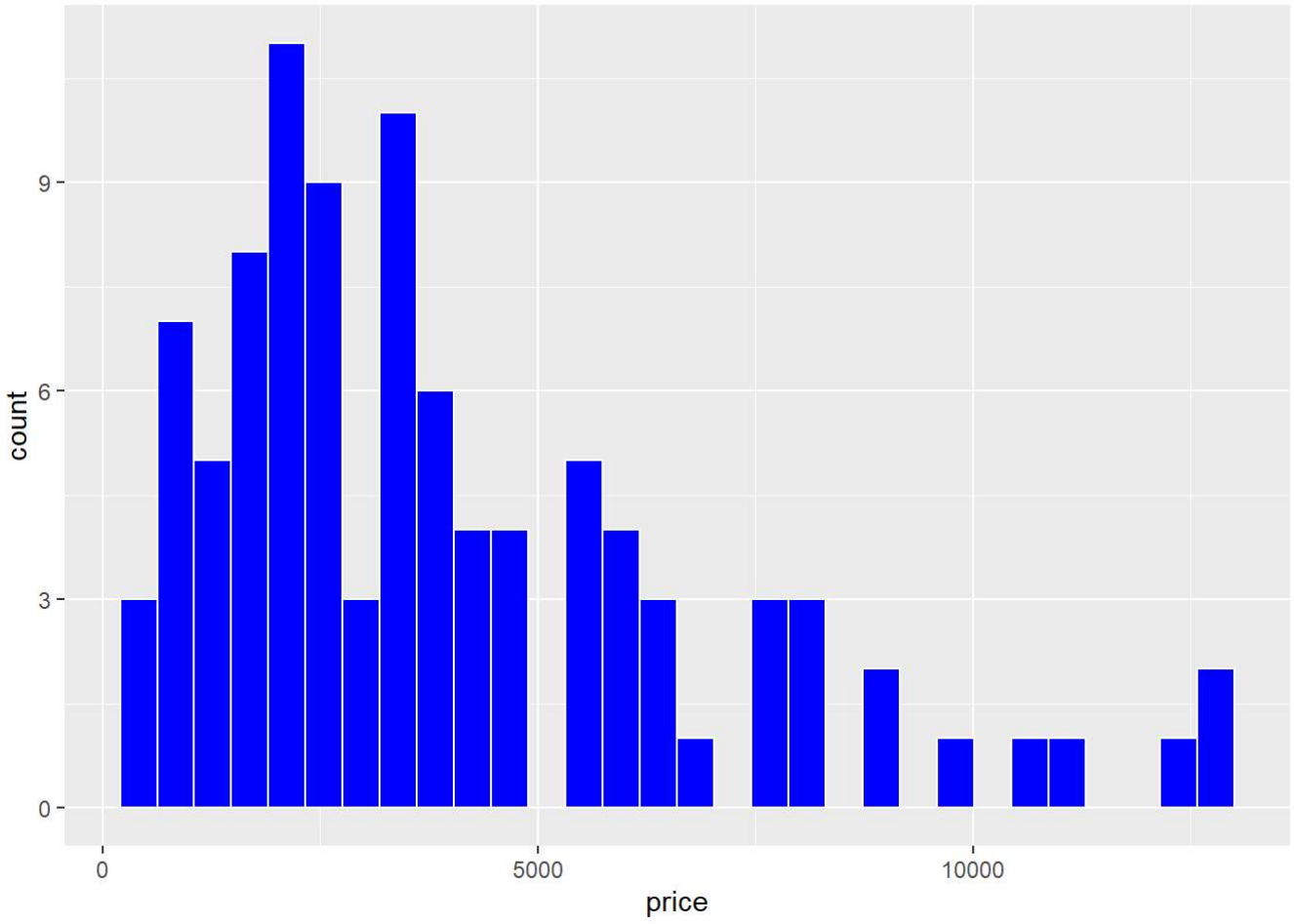


12. More visualizations (scatter, histogram, density, boxplot, labels, lollipop, heatmap)

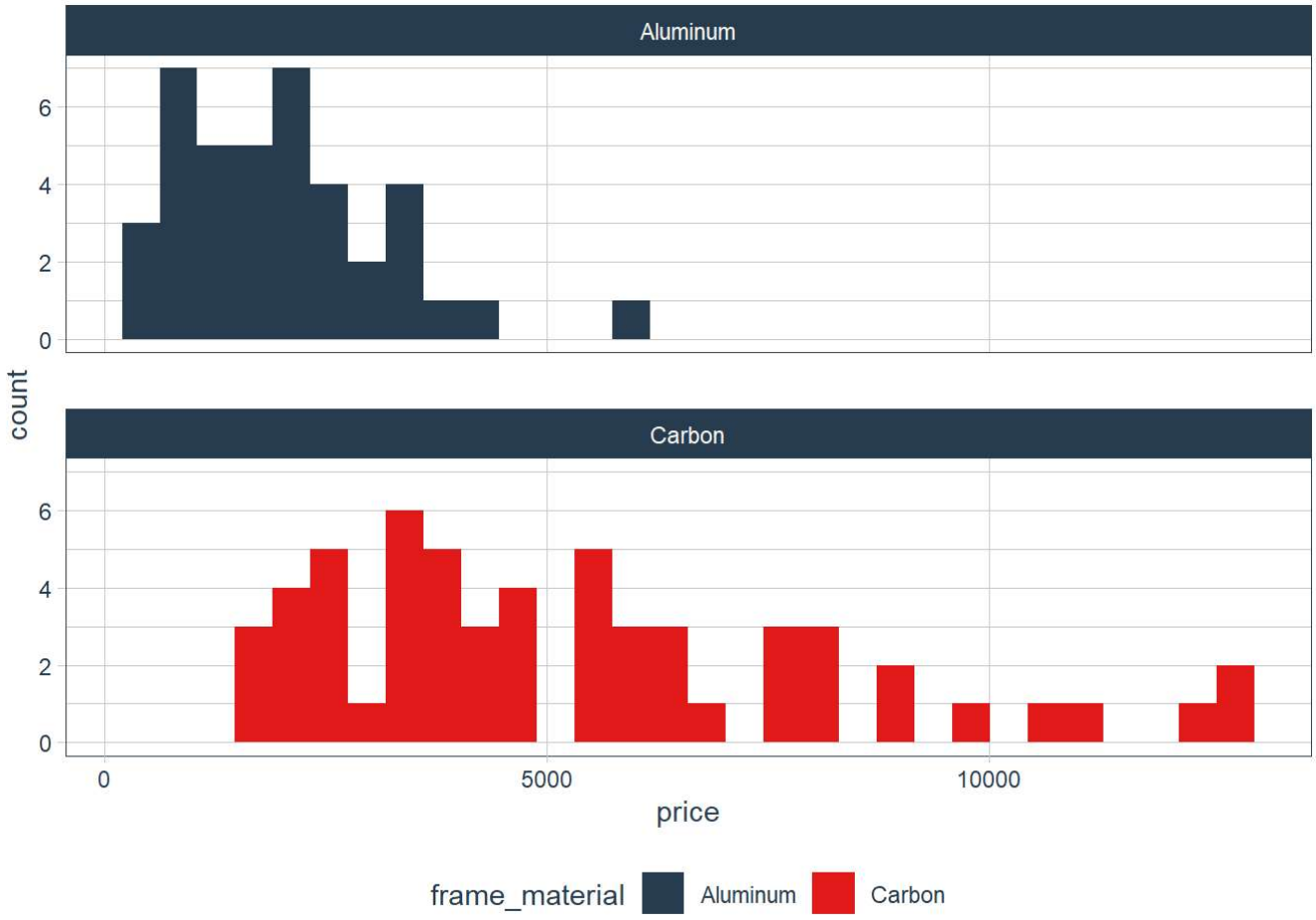
```
# Scatter: order value
order_value_tbl <- bike_orderlines_wrangled_tbl %>%
  select(order_id, order_line, total_price, quantity) %>%
  group_by(order_id) %>%
  summarize(
    total_quantity = sum(quantity),
    total_price    = sum(total_price)
  ) %>%
  ungroup()

order_value_tbl %>%
  ggplot(aes(x = total_quantity, y = total_price)) +
  geom_point(alpha = 0.5, size = 2)
```



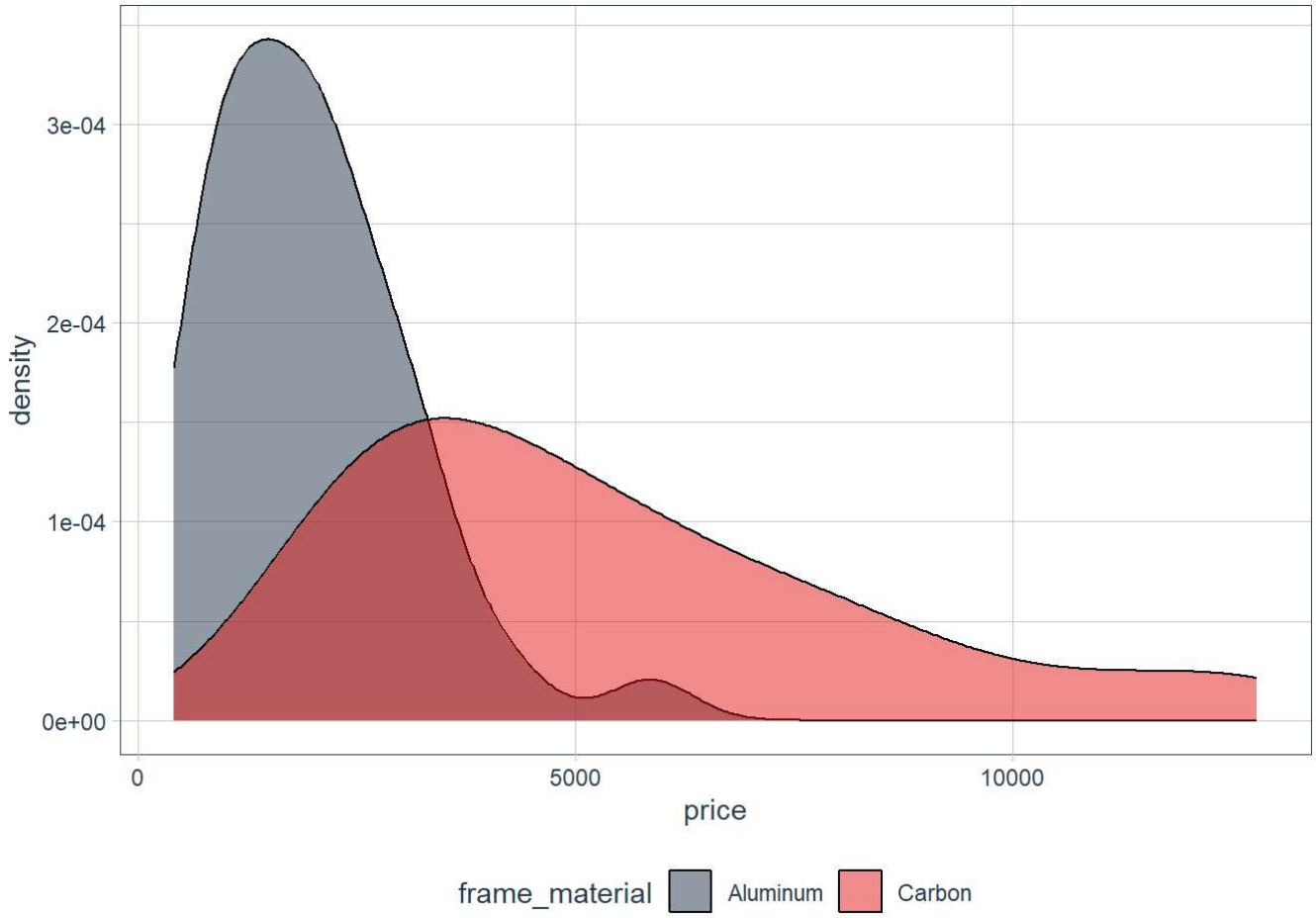

```
# Histogram / density
bike_orderlines_wrangled_tbl %>%
  distinct(model, price) %>%
  ggplot(aes(price)) +
  geom_histogram(bins = 30, fill = "blue", color = "white")
```



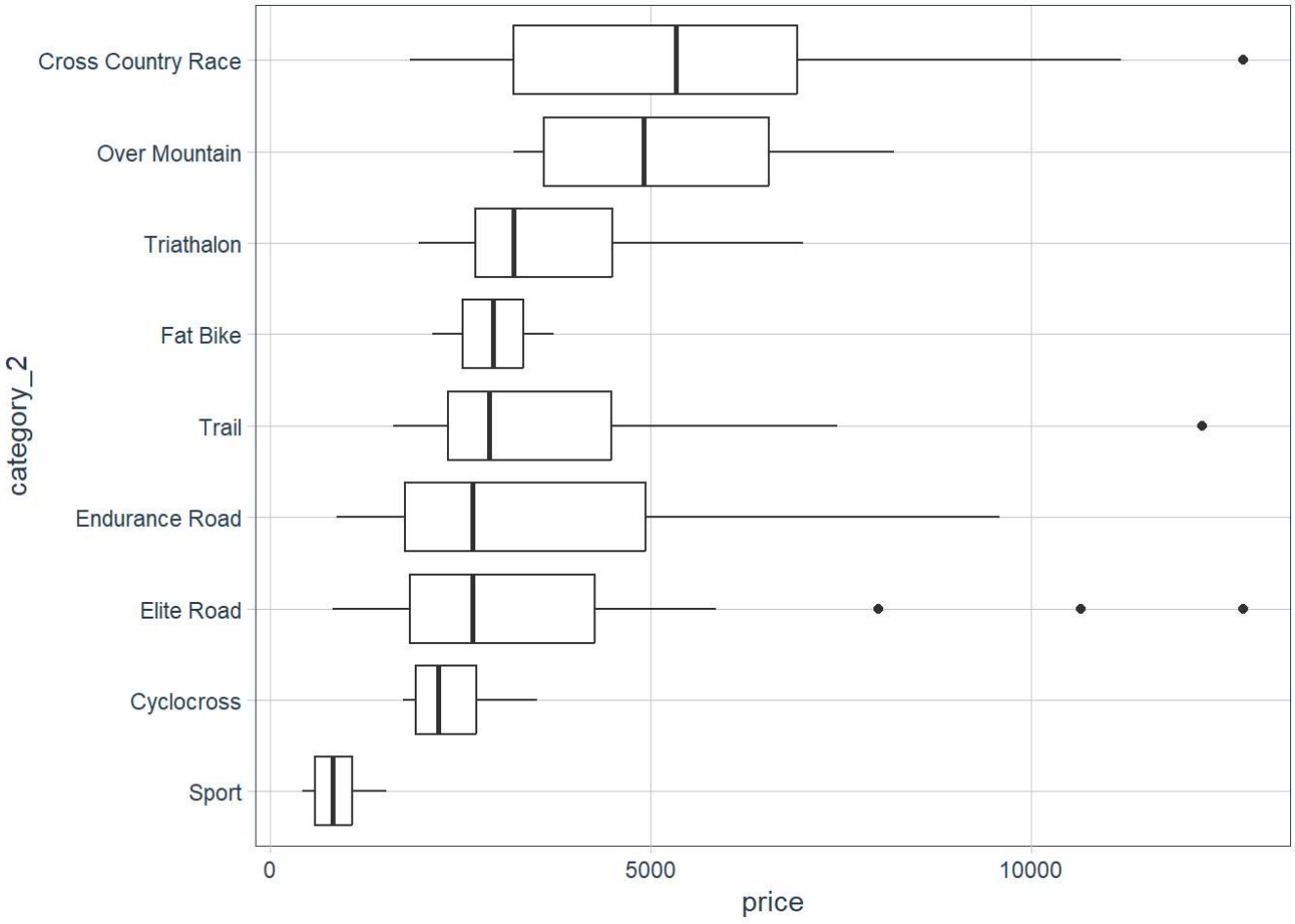

```
bike_orderlines_wrangled_tbl %>%
  distinct(price, model, frame_material) %>%
  ggplot(aes(price, fill = frame_material)) +
  geom_histogram() +
  facet_wrap(~ frame_material, ncol = 1) +
  scale_fill_tq() +
  theme_tq()
```




```
# Density
bike_orderlines_wrangled_tbl %>%
  distinct(price, model, frame_material) %>%
  ggplot(aes(price, fill = frame_material)) +
  geom_density(alpha = 0.5)+
  scale_fill_tq() +
  theme_tq()
```

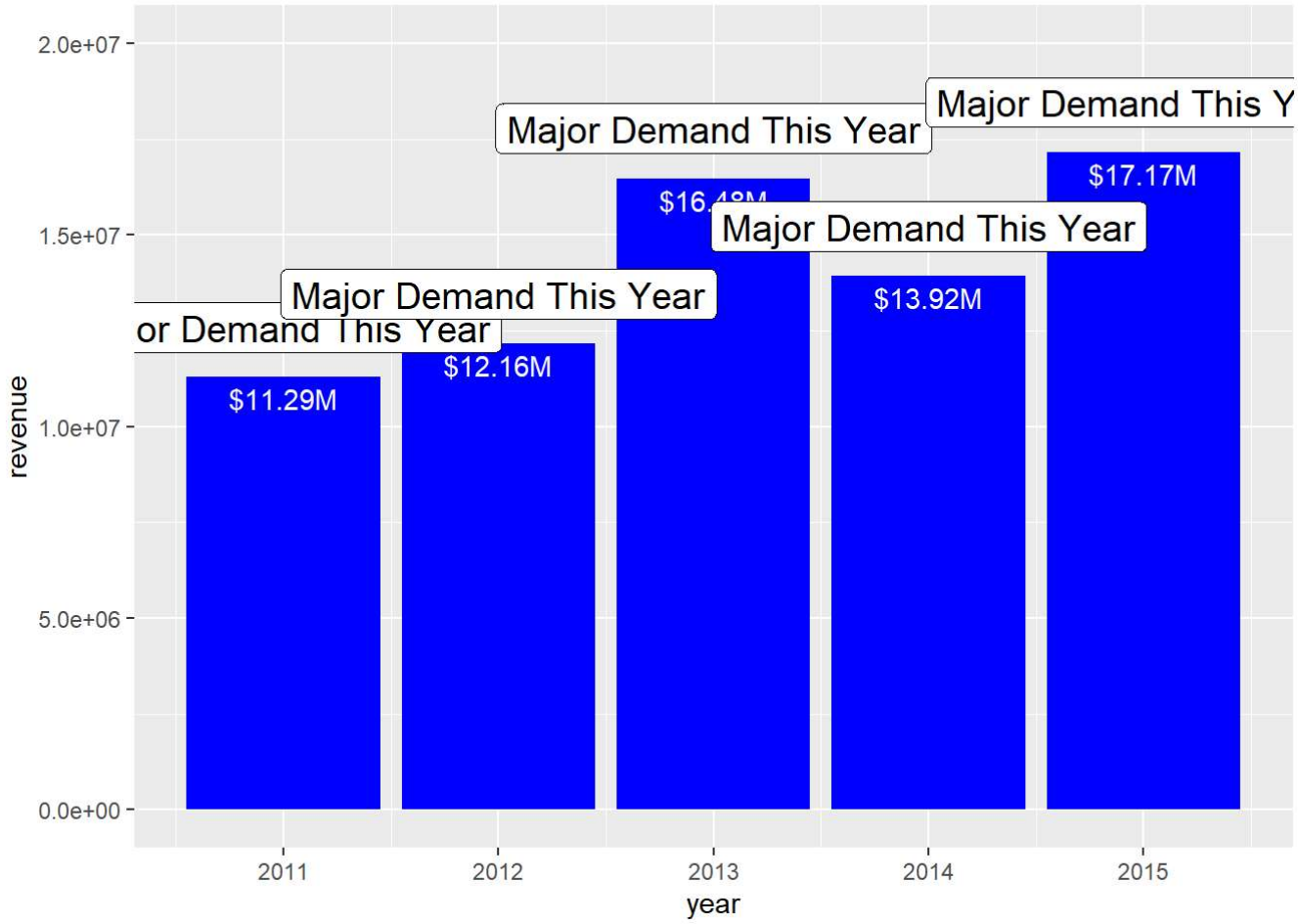



```
# Box plot
unit_price_by_cat2_tbl <- bike_orderlines_wrangled_tbl %>%
  select(category_2, model, price) %>%
  distinct() %>%
  mutate(category_2 = as_factor(category_2) %>% fct_reorder(price))

unit_price_by_cat2_tbl %>%
  ggplot(aes(category_2, price)) +
  geom_boxplot() +
  coord_flip() +
  theme_tq()
```



13. Adding text and labels to plots

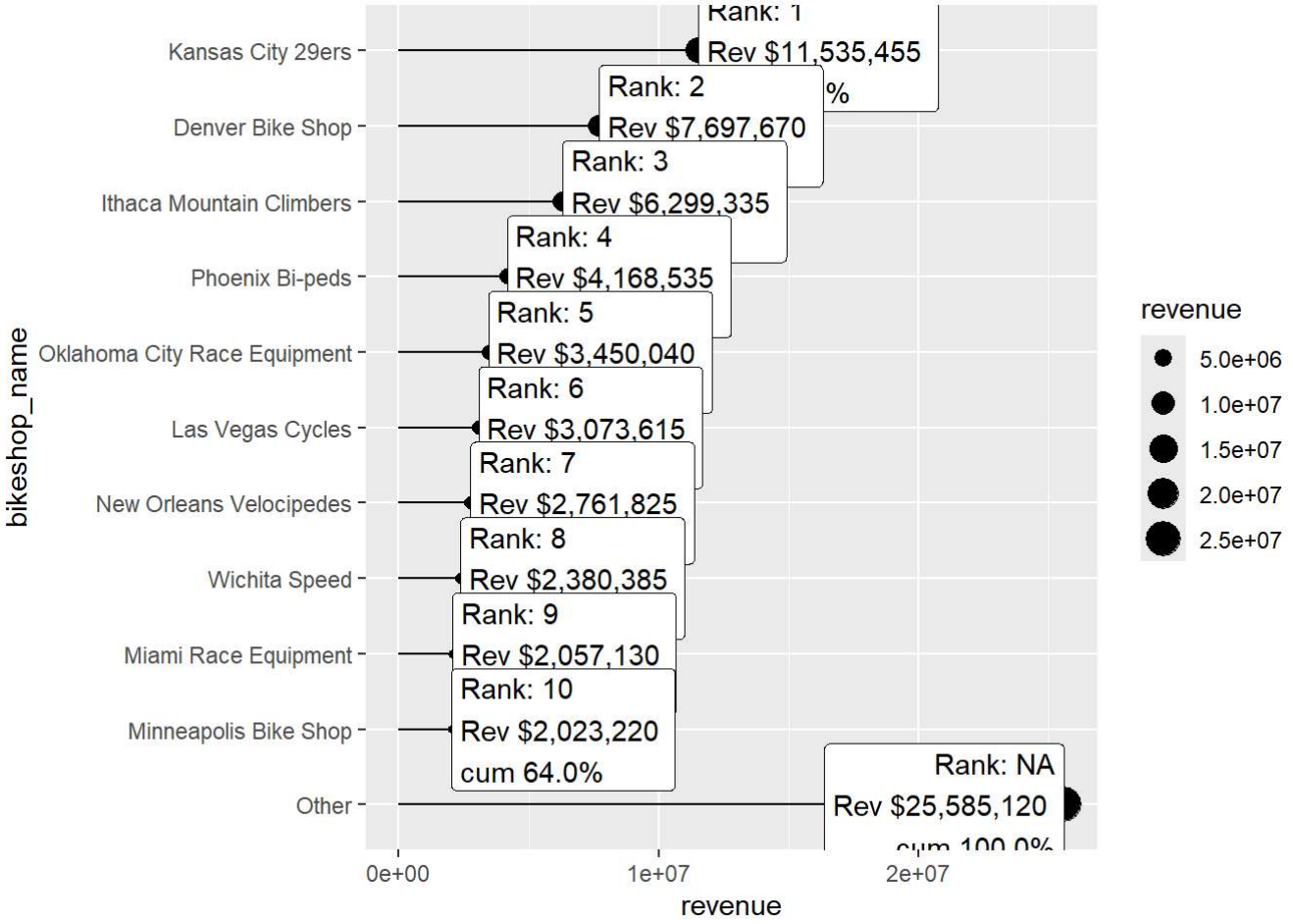
```
revenue_by_year <- bike_orderlines_wrangled_tbl %>%
  select(order_date, total_price) %>%
  mutate(year = year(order_date)) %>%
  group_by(year) %>%
  summarize(revenue = sum(total_price)) %>%
  ungroup()

revenue_by_year %>%
  ggplot(aes(year, revenue)) +
  geom_col(fill = "blue") +
  geom_text(aes(label = scales::dollar(revenue, scale = 1e-6, suffix = "M")),
            vjust = 1.5, color = "white") +
  geom_label(label = "Major Demand This Year",
             vjust = -0.5,
             size = 5) +
  expand_limits(y = 2e7)
```



14. Top N customers (lollipop) & heatmaps

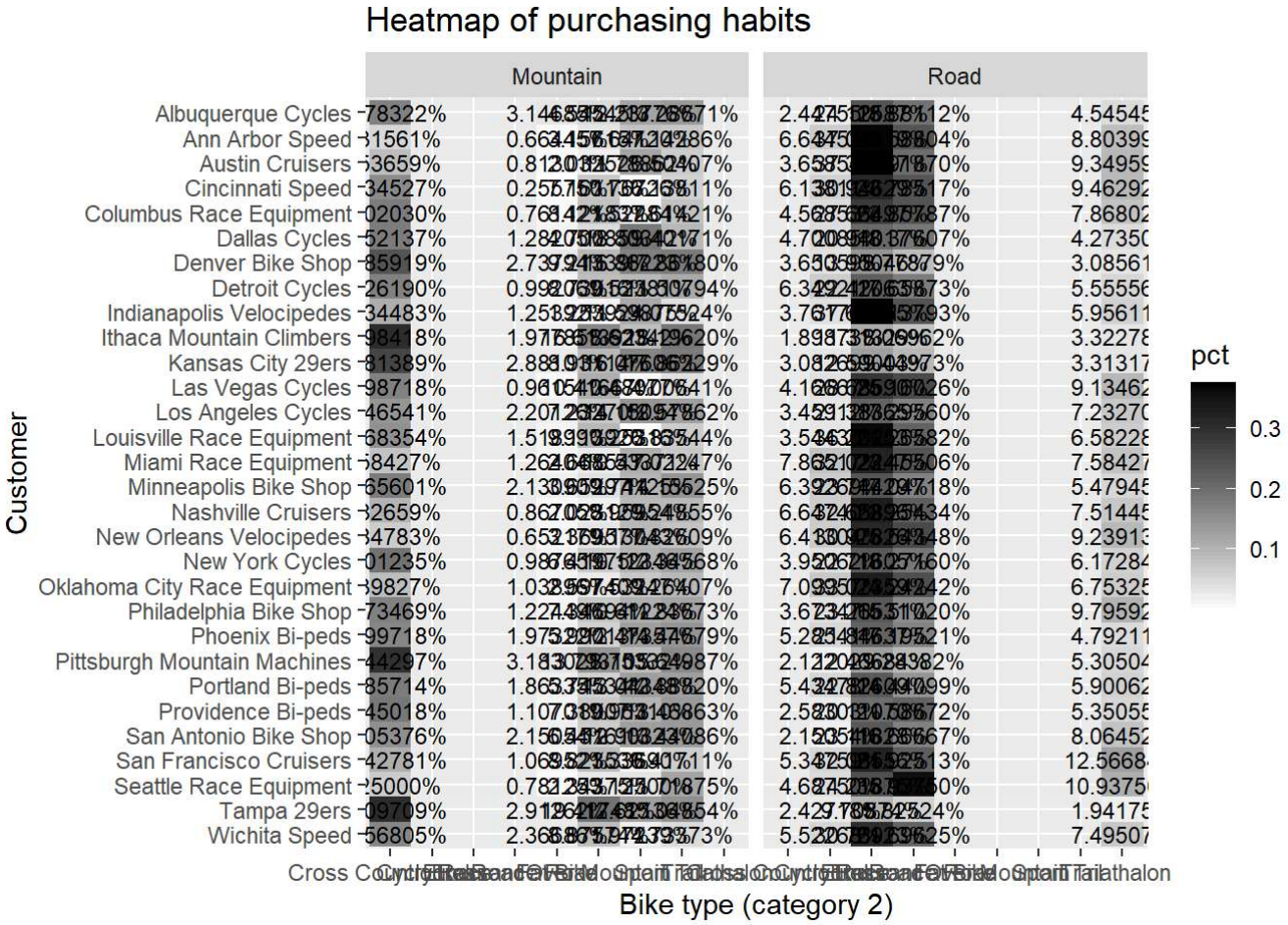
```
n <- 10

topN_customers <- bike_orderlines_wrangled_tbl %>%
  select(bikeshop_name, total_price) %>%
  mutate(bikeshop_name = as_factor(bikeshop_name) %>% fct_lump(n = n, w = total_price)) %>%
  group_by(bikeshop_name) %>%
  summarize(revenue = sum(total_price)) %>%
  ungroup() %>%
  mutate(bikeshop_name = bikeshop_name %>% fct_reorder(revenue)) %>%
  mutate(bikeshop_name = bikeshop_name %>% fct_relevel("Other", after =0)) %>%
  arrange(desc(bikeshop_name)) %>%
  mutate(revenue_text = scales::dollar(revenue)) %>%
  mutate(cum_pct = cumsum(revenue) / sum(revenue)) %>%
  mutate(cum_pct_txt = scales::percent(cum_pct)) %>%
  mutate(rank = row_number()) %>%
  mutate(rank = case_when(
    rank == max(rank) ~NA_integer_,
    TRUE ~ rank
  )) %>%
  mutate(label_text = str_glue("Rank: {rank}\nRev {revenue_text} \ncum {cum_pct_txt}"))
```

```
topN_customers %>%
  ggplot(aes(revenue, bikeshop_name)) +
  geom_segment(aes(xend = 0, yend = bikeshop_name)) +
  geom_point(aes(size = revenue)) +
  geom_label(aes(label = label_text), hjust = "inward")
```



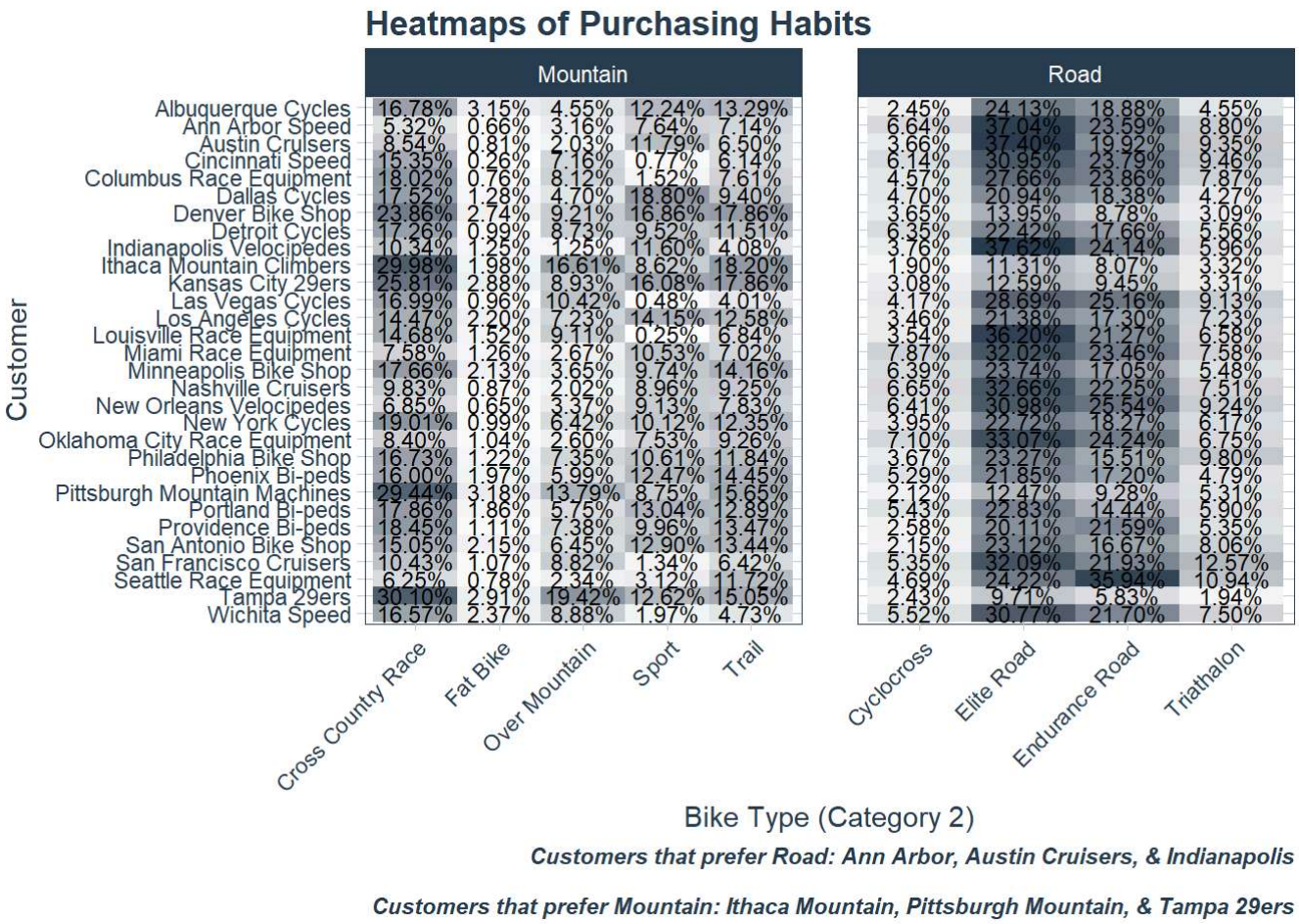

```
# Heatmap of purchasing habits
pct_sales_by_customer_tbl <- bike_orderlines_wrangled_tbl %>%
  select(bikeshop_name, category_1, category_2, quantity) %>%
  group_by(bikeshop_name, category_1, category_2) %>%
  summarise(total_qty = sum(quantity)) %>%
  ungroup() %>%
  group_by(bikeshop_name) %>%
  mutate(pct = total_qty / sum(total_qty)) %>%
  ungroup() %>%
  mutate(bikeshop_name  = as.factor(bikeshop_name) %>% fct_rev()) %>%
  mutate(bikeshop_name_num = as.numeric(bikeshop_name))

pct_sales_by_customer_tbl %>%
  ggplot(aes(category_2, bikeshop_name)) +
  geom_tile(aes(fill = pct)) +
  geom_text(aes(label = scales::percent(pct)), size = 3) +
```

```
  scale_fill_gradient(low = "white", high = "black") +
  facet_wrap(~ category_1) +
  labs(
    title = "Heatmap of purchasing habits",
    x     = "Bike type (category 2)",
    y     = "Customer"
  )
```




```
pct_sales_by_customer_tbl %>%
  ggplot(aes(category_2, bikeshop_name)) +
  geom_tile(aes(fill = pct)) +
  geom_text(aes(label = scales::percent(pct, accuracy = 0.01L)), size = 3) +
  facet_wrap(~ category_1, scales = "free_x") +
  scale_fill_gradient(low = "white", high = palette_light()[1]) +
  labs(
    title = "Heatmaps of Purchasing Habits",
    x     = "Bike Type (Category 2)",
    y     = "Customer",
    caption = str_glue(
      "Customers that prefer Road: Ann Arbor, Austin Cruisers, & Indianapolis

       Customers that prefer Mountain: Ithaca Mountain, Pittsburgh Mountain, & Tampa 29ers")
  ) +
```

```
  theme_tq() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    legend.position = "none",
    plot.title = element_text(face = "bold"),
    plot.caption = element_text(face = "bold.italic")
  )
```



15. String manipulation & feature engineering

```
# str_detect()
bikes_tbl %>%
  select(model) %>%
  mutate(supersix = model %>% str_detect("Supersix") %>% as.numeric()) %>%
  mutate(black    = model %>% str_detect("Black") %>% as.numeric())
```

```
# A tibble: 97 × 3
   model                          supersix black
   <chr>                             <dbl> <dbl>
 1 Supersix Evo Black Inc.               1     1
 2 Supersix Evo Hi-Mod Team              1     0
 3 Supersix Evo Hi-Mod Dura Ace 1        1     0
```

```
 4 Supersix Evo Hi-Mod Dura Ace 2        1     0
 5 Supersix Evo Hi-Mod Utegra            1     0
 6 Supersix Evo Red                      1     0
 7 Supersix Evo Ultegra 3                1     0
 8 Supersix Evo Ultegra 4                1     0
 9 Supersix Evo 105                      1     0
10 Supersix Evo Tiagra                   1     0
# ℹ 87 more rows
```

```
# Concatenation
order_id <- 1
order_line <- 1
str_c("Order Line: ", order_id, ".", order_line)
```

```
[1] "Order Line: 1.1"
```

```
# str_glue
str_glue("Order Line: {order_id}.{order_line}")
```

```
Order Line: 1.1
```

```
bike_orderlines_tbl %>%
  select(bikeshop_name, order_id, order_line) %>%
  mutate(purchase_statement = str_glue(
    "Order Line: {order_id}.{order_line} sent to Customer: {str_to_upper(bikeshop_name)}"
  ) %>% as.character())
```

```
# A tibble: 15,644 × 4
   bikeshop_name             order_id order_line purchase_statement
   <chr>                        <dbl>      <dbl> <chr>
 1 Ithaca Mountain Climbers         1          1 Order Line: 1.1 sent to Custom…
 2 Ithaca Mountain Climbers         1          2 Order Line: 1.2 sent to Custom…
 3 Kansas City 29ers                2          1 Order Line: 2.1 sent to Custom…
 4 Kansas City 29ers                2          2 Order Line: 2.2 sent to Custom…
 5 Louisville Race Equipment        3          1 Order Line: 3.1 sent to Custom…
 6 Louisville Race Equipment        3          2 Order Line: 3.2 sent to Custom…
 7 Louisville Race Equipment        3          3 Order Line: 3.3 sent to Custom…
 8 Louisville Race Equipment        3          4 Order Line: 3.4 sent to Custom…
 9 Louisville Race Equipment        3          5 Order Line: 3.5 sent to Custom…
10 Ann Arbor Speed                  4          1 Order Line: 4.1 sent to Custom…
# ℹ 15,634 more rows
```

```
# Separating text
bikes_tbl %>% select(description) %>%
  separate(col = description,
           into = c("category_1", "category_2", "frame_material"),
           sep  = " - ",
           remove = FALSE)
```

```
# A tibble: 97 × 4
   description                category_1 category_2 frame_material
   <chr>                      <chr>      <chr>      <chr>
 1 Road - Elite Road - Carbon Road       Elite Road Carbon
 2 Road - Elite Road - Carbon Road       Elite Road Carbon
 3 Road - Elite Road - Carbon Road       Elite Road Carbon
 4 Road - Elite Road - Carbon Road       Elite Road Carbon
 5 Road - Elite Road - Carbon Road       Elite Road Carbon
 6 Road - Elite Road - Carbon Road       Elite Road Carbon
 7 Road - Elite Road - Carbon Road       Elite Road Carbon
 8 Road - Elite Road - Carbon Road       Elite Road Carbon
 9 Road - Elite Road - Carbon Road       Elite Road Carbon
10 Road - Elite Road - Carbon Road       Elite Road Carbon
# ℹ 87 more rows
```

16. Feature engineering: model parsing example

This section shows how to clean model names and extract base and tier information.

```
test <- bikes_tbl %>% select(model) %>%
  # fix typo
  mutate(model = case_when(
    model == "CAAD Disc Ultegra" ~ "CAAD12 Disc Ultegra",
    model == "Supersix Evo Hi-Mod Utegra" ~ "Supersix Evo Hi-Mod Ultegra",
    model == "Syapse Carbon Tiagra" ~ "Synapse Carbon Tiagra",
    TRUE ~ model)
  ) %>%
  # separating using spaces
  separate(col = model,
           into = str_c("model_", 1:7),
           sep = " ",
           remove = FALSE,
           fill = "right"
  ) %>%
  mutate(model_base = case_when(
    # Fix Supersix Evo
    str_detect(str_to_lower(model_1), "supersix") ~ 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 Fat CAAD Bikes
    str_detect(str_to_lower(model_1), "fat") ~ 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 = " "),
    # 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 using str_detect()
  mutate(black  = model_tier %>% str_to_lower() %>% str_detect("black") %>% as.numeric(),
         red    = model_tier %>% str_to_lower() %>% str_detect("red") %>% 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(),
         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())
```

17. Reusable function: separate_bike_model

```
data <- bikes_tbl

separate_bike_model <- function(data, append = TRUE) {
  # If append == FALSE, only keep model column
  if (!append){
    data <- data %>% select(model)
  }

  output_tbl <- data %>% select(model) %>%
    # fix typos
    mutate(model = case_when(
      model == "CAAD Disc Ultegra" ~ "CAAD12 Disc Ultegra",
      model == "Supersix Evo Hi-Mod Utegra" ~ "Supersix Evo Hi-Mod Ultegra",
      model == "Syapse Carbon Tiagra" ~ "Synapse Carbon Tiagra",
      TRUE ~ model)
    ) %>%
    # separate using spaces
    separate(col = model,
             into = str_c("model_", 1:7),
             sep = " ",
             remove = FALSE,
             fill = "right"
    ) %>%
    mutate(model_base = case_when(
      str_detect(str_to_lower(model_1), "supersix") ~ str_c(model_1, model_2, sep = " "),
      str_detect(str_to_lower(model_1), "beast") ~ str_c(model_1, model_2, model_3, model_4, sep =
      str_detect(str_to_lower(model_1), "bad") ~ str_c(model_1, model_2, sep = " "),
      str_detect(str_to_lower(model_1), "fat") ~ str_c(model_1, model_2, sep = " "),
      str_detect(str_to_lower(model_1), "29") ~ str_c(model_1, model_2, sep = " "),
      TRUE ~ model_1
    )) %>%
    mutate(model_tier = model %>% str_replace(model_base, replacement = "") %>% str_trim()) %>%
    select(-matches("model_[0-9]")) %>%
    mutate(black  = model_tier %>% str_to_lower() %>% str_detect("black") %>% as.numeric(),
```

```
            red    = model_tier %>% str_to_lower() %>% str_detect("red") %>% 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(),
            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())

  return(output_tbl)
}

separate_bike_model(data)
```

```
# A tibble: 97 × 10
   model   model_base model_tier black   red hi_mod  team ultegra dura_ace  disc
   <chr>   <chr>      <chr>      <dbl> <dbl>  <dbl> <dbl>   <dbl>    <dbl> <dbl>
 1 Supers… Supersix … Black Inc.     1     0      0     0       0        0     0
 2 Supers… Supersix … Hi-Mod Te…     0     0      0     1       0        0     0
 3 Supers… Supersix … Hi-Mod Du…     0     0      0     0       0        0     0
 4 Supers… Supersix … Hi-Mod Du…     0     0      0     0       0        0     0
 5 Supers… Supersix … Hi-Mod Ul…     0     0      0     0       1        0     0
 6 Supers… Supersix … Red            0     1      0     0       0        0     0
 7 Supers… Supersix … Ultegra 3      0     0      0     0       1        0     0
 8 Supers… Supersix … Ultegra 4      0     0      0     0       1        0     0
 9 Supers… Supersix … 105            0     0      0     0       0        0     0
10 Supers… Supersix … Tiagra         0     0      0     0       0        0     0
# ℹ 87 more rows
```