Bikes categoty price prediction

7/24/2020

Contents

Challenge Summary

#Which Bike Categories are in high demand? #Which Bike Categories are under represented?

Goal

#Use a pricing algorithm to determine a new product price in a category gap

Procedure

- 1.Get the ingredients **(recipe())**: specify the response variable and predictor variables
- 2.Write the recipe **(step_xxx())**: define the pre-processing steps, such as imputation
- 3.creating **dummy variables**, scaling, and more
- 4.Prepare the recipe **(prep())**: provide a dataset to base each step on
- 5.Bake the recipe **(bake())**: apply the pre-processing steps to your datasets
- 6.Create a workflow **(workflow())**: Add models add_model() and the recipe add_recipe()
- 7.Predict the price of a new model **fit()** and **predict()**

Libraries

Load the following libraries.

```
# install.packages("plotly")
# Standard
library(tidyverse)
# Modeling
library(parsnip)
# Preprocessing & Sampling
library(recipes)
library(rsample)
# Modeling Error Metrics
library(yardstick)
# Plotting Decision Trees
library(rpart.plot)
library(tidymodels)  # for the parsnip package, along with the rest of tidymodels
# Helper packages
library(broom.mixed) # for converting bayesian models to tidy tibbles
library(rstanarm)
library(dials)
library(workflows)
library(vip)
library(janitor)
```

Data

We will be using bike features data table

- `price`: The target element, we want to predict bike price correctly
- `category`: Bikes family which will be studied
- `bike components manufacturere`: Components which will help predicting the price depending on the manufacturer
- `Weight`: Component which is also involved during price prediction.
- `Model`: Bike model, will be considered as an ID for a certain bike.
- `Frame material`: will be used as Carbon by default

```
# Bike features
bike_features_tbl <- readRDS("D:/Mechatronics_master/Third semester/Data science/Machine learning/ml_jou
glimpse(bike_features_tbl)
```

```
## Rows: 231
## Columns: 67
## $ bike_id                     <dbl> 2875, 2873, 2874, 2876, 2877, 2225, 209...
## $ model                       <chr> "Aeroad CF SL Disc 8.0 Di2", "Aeroad CF...
## $ model_year                  <dbl> 2020, 2020, 2020, 2020, 2020, 2019, 201...
## $ frame_material              <chr> "carbon", "carbon", "carbon", "carbon",...
## $ weight                      <dbl> 7.60, 7.27, 7.10, 7.73, 7.83, 6.80, 6.8...
## $ price                       <dbl> 4579, 6919, 6429, 5069, 3609, 6139, 535...
## $ category_1                  <chr> "Road", "Road", "Road", "Road", "Road",...
## $ category_2                  <chr> "Race", "Race", "Race", "Race", "Race",...
## $ category_3                  <chr> "Aeroad", "Aeroad", "Aeroad", "Aeroad",...
## $ gender                      <chr> "unisex", "unisex", "unisex", "unisex",...
## $ url                         <chr> "https://www.canyon.com/en-de/road-bike...
## $ Frame                       <chr> "Canyon Aeroad CF SL Disc", "Canyon Aer...
## $ Fork                        <chr> "Canyon FK0041 CF SLX Disc", "Canyon FK...
## $ ‘Rear Derailleur‘           <chr> "Shimano Ultegra Di2 R8050 SS", "SRAM R...
## $ ‘Front Derailleur‘          <chr> "Shimano Ultegra Di2 R8050", "SRAM RED ...
## $ Cassette                    <chr> "Shimano Ultegra R8000, 11-speed, 11-28...
## $ Crank                       <chr> "Shimano Ultegra R8000", "SRAM RED D1",...
## $ ‘Bottom bracket‘            <chr> "Shimano Pressfit BB72", "SRAM Pressfit...
## $ ‘Thru Axle‘                 <chr> "Canyon Thru Axle", "Canyon Thru Axle",...
## $ Cockpit                     <chr> "Canyon H36 Aerocockpit CF", "Canyon H3...
## $ Saddle                      <chr> "Selle Italia SLR", "Selle Italia SLR",...
## $ Seatpost                    <chr> "Canyon S27 Aero VCLS CF", "Canyon S27 ...
## $ Pedals                      <chr> "None included", "None included", "None...
## $ ‘Derailleur hanger‘         <chr> "Shop Derailleur Hanger GP0211-01", "Sh...
## $ Battery                     <chr> "", "SRAM eTap Powerpack", "", "SRAM eT...
## $ Brake                       <chr> "", "", "", "", "", "", "", "", "", "",...
## $ ‘Shift Lever‘               <chr> "", "", "", "", "", "", "Shimano Di2 Re...
## $ Chain                       <chr> "", "", "", "", "", "", "Shimano CN-HG9...
## $ Stem                        <chr> "", "", "", "", "", "", "", "Canyon V13...
## $ Handlebar                   <chr> "", "", "", "", "", "", "", "Canyon H16...
## $ Headset                     <chr> "", "", "", "", "", "", "", "", "", "",...
## $ Motor                       <chr> "", "", "", "", "", "", "", "", "", "",...
## $ ‘Battery Charger‘           <chr> "", "", "", "", "", "", "", "", "", "",...
## $ ‘Flat Pedals‘               <chr> "", "", "", "", "", "", "", "", "", "",...
## $ Chainguard                  <chr> "", "", "", "", "", "", "", "", "", "",...
## $ ‘Aero Bar‘                  <chr> "", "", "", "", "", "", "", "", "", "",...
## $ ‘Brake Lever / Master‘      <chr> "", "", "", "", "", "", "", "", "", "",...
## $ ‘Wheel Tire System‘         <chr> "", "", "", "", "", "", "", "", "", "",...
## $ ‘Suspension Fork‘           <chr> "", "", "", "", "", "", "", "", "", "",...
## $ ‘Disc Brake‘                <chr> "", "", "", "", "", "", "", "", "", "",...
## $ Grips                       <chr> "", "", "", "", "", "", "", "", "", "",...
## $ Chainring                   <chr> "", "", "", "", "", "", "", "", "", "",...
## $ Display                     <chr> "", "", "", "", "", "", "", "", "", "",...
## $ Modeswitch                  <chr> "", "", "", "", "", "", "", "", "", "",...
## $ ‘Rear Shock‘                <chr> "", "", "", "", "", "", "", "", "", "",...
## $ Light                       <chr> "", "", "", "", "", "", "", "", "", "",...
## $ Fender                      <chr> "", "", "", "", "", "", "", "", "", "",...
## $ ‘Bike Racks‘                <chr> "", "", "", "", "", "", "", "", "", "",...
## $ ‘Brake 1‘                   <chr> "", "", "", "", "", "SRAM S-900 Direct ...
## $ ‘Brake 2‘                   <chr> "", "", "", "", "", "SRAM S-900 Direct ...
## $ ‘Shift-/ Brake Lever 1‘     <chr> "Shimano Ultegra Di2 R8070, 11-speed", ...
## $ ‘Shift-/ Brake Lever 2‘     <chr> "Shimano Ultegra Di2 R8070, 11-speed", ...
## $ ‘Wheel 1‘                   <chr> "DT Swiss ARC 1400 Dicut", "DT Swiss AR...
## $ ‘Wheel 2‘                   <chr> "DT Swiss ARC 1400 Dicut", "DT Swiss AR...
## $ ‘Tyre 1‘                    <chr> "Continental Grand Prix 5000 / Attack  ...
## $ ‘Tyre 2‘                    <chr> "Continental Grand Prix 5000, 25 mm", "...
## $ ‘Handlebar Tape 1‘          <chr> "Canyon Ergospeed Gel", "Canyon Ergospe...
## $ ‘Handlebar Tape 2‘          <chr> "Canyon bar-end plug", "Canyon bar-end ...
## $ ‘Manuals and Accessories 1‘ <chr> "Canyon tool case", "Canyon tool case",...
## $ ‘Manuals and Accessories 2‘ <chr> "DT Swiss warranty & intended use manua...
## $ ‘Manuals and Accessories 3‘ <chr> "Canyon starter box", "Canyon starter b...
## $ ‘Manuals and Accessories 4‘ <chr> "", "", "", "", "", "", "", "", "", "BA...
## $ ‘Manuals and Accessories 5‘ <chr> "", "", "", "", "", "", "", "", "", "",...
## $ ‘Manuals and Accessories 6‘ <chr> "", "", "", "", "", "", "", "", "", "",...
## $ ‘Manuals and Accessories 7‘ <chr> "", "", "", "", "", "", "", "", "", "",...
## $ ‘Manuals and Accessories 8‘ <chr> "", "", "", "", "", "", "", "", "", "",...
## $ ‘Brake Rotor‘                <list> ["Shimano RT800", "SRAM Centerline X",...
```

Step 1 - re-organize the data set.

```
#Define the category which shall be analyzed
category <- "category_2"
predictors_filter = 7
```

```
# Apply your data transformation skills!
bike_features_tbl_r <- bike_features_tbl %>%
                       select(price, model:weight , category,‘Rear Derailleur‘, ‘Saddle‘, ‘Shift Lever‘
                       mutate_all(funs(replace(., .=="", "N/A"))) %>%
                       mutate(id = row_number())%>%
                       mutate(id_1 = row_number())%>%
                       mutate(id_2 = row_number())%>%
                       mutate(id_3 = row_number())%>%
                       pivot_wider(names_from = ‘Saddle‘, values_from =id_3, names_repair = "unique") %
                       pivot_wider(names_from = ‘Shift Lever‘, values_from =id_2, names_repair = "uniqu
                       pivot_wider(names_from = ‘Rear Derailleur‘, values_from =id_1, names_repair = "u
                       select(-‘N/A...59‘, -‘N/A...60‘, -‘N/A...133‘ ) %>%
                       mutate_all(funs(replace_na(.,0)))

bike_features_tbl_r$price <- as.integer(bike_features_tbl_r$price)
bike_features_tbl_r$weight <- as.double(bike_features_tbl_r$weight)

bike_features_tbl_r_t <-bike_features_tbl_r %>%
                       select(-(price:id)) %>%
                       mutate_all(funs(replace(.,.>1,1))) %>%
                       mutate_if(is.character,as.numeric) %>%
                       add_column(test_col = "Don’t care", .before = TRUE) %>%
                       adorn_totals(where ="row",name = "Total") %>%
                       select(-test_col)


ff = data.frame(dummy =1:231)
for(i in 1:ncol(bike_features_tbl_r_t)) {       # for-loop over columns

  if (bike_features_tbl_r_t[232,i] >= predictors_filter)
  {
    ff <- ff%>% add_column(bike_features_tbl_r_t[1:231,i],.after = TRUE)
  }
}

bike_features_tbl_r_t <- ff %>% select(-dummy)

bike_features_tbl_r_tt <- bike_features_tbl_r_t  %>% add_column(.data = bike_features_tbl_r %>%
                                                                  select(price:id))
bike_features_tbl_r_tt
```

```
## # A tibble: 231 x 36
##    price model model_year frame_material weight category_2    id
##    <int> <chr> <chr>      <chr>           <dbl> <chr>      <dbl>
##  1  4579 Aero~ 2020       carbon           7.6  Race           1
##  2  6919 Aero~ 2020       carbon           7.27 Race           2
##  3  6429 Aero~ 2020       carbon           7.1  Race           3
##  4  5069 Aero~ 2020       carbon           7.73 Race           4
##  5  3609 Aero~ 2020       carbon           7.83 Race           5
##  6  6139 Aero~ 2019       carbon           6.8  Race           6
##  7  5359 Aero~ 2019       carbon           6.8  Race           7
##  8  2629 Aero~ 2021       carbon           7.6  Race           8
##  9  3699 Aero~ 2020       carbon           7.3  Race           9
## 10  3219 Aero~ 2020       carbon           7.2  Race          10
## # ... with 221 more rows, and 29 more variables: ‘SRAM X01 Eagle‘ <dbl>,
## #   ‘Shimano Deore XT‘ <dbl>, ‘Shimano Deore XTR‘ <dbl>, ‘SRAM GX Eagle‘ <dbl>,
## #   ‘Shimano 105 R7000 GS‘ <dbl>, ‘Shimano Ultegra R8000 SGS‘ <dbl>, ‘Shimano
## #   Ultegra R8000 SS‘ <dbl>, ‘Shimano Dura-Ace Di2 R9150, 11-speed‘ <dbl>,
## #   ‘Shimano Ultegra Di2 R8050 SS‘ <dbl>, ‘Shimano 105 RS700 2s‘ <dbl>, ‘SRAM
## #   X01 Eagle Trigger 12s‘ <dbl>, ‘Shimano Deore XT, 12-speed‘ <dbl>, ‘Shimano
## #   Deore M6100 12s‘ <dbl>, ‘Shimano SLX M7100 12s‘ <dbl>, ‘Shimano Deore XTR,
## #   12-speed‘ <dbl>, ‘SRAM NX Eagle Trigger 12s‘ <dbl>, ‘Fizik Essenza‘ <dbl>,
## #   ‘Canyon Sport Saddle EP1249‘ <dbl>, ‘Canyon SD:ON‘ <dbl>, ‘Iridium
## #   Trail‘ <dbl>, ‘Fizik Antares R5‘ <dbl>, ‘Fizik Antares R3‘ <dbl>, ‘Fizik
## #   Mistica‘ <dbl>, ‘Selle Italia Model X‘ <dbl>, ‘Iridium Fitness‘ <dbl>,
## #   ‘Fizik Aliante R5‘ <dbl>, ‘Selle Italia X3 Lady‘ <dbl>, ‘Selle Italia
## #   X3‘ <dbl>, ‘Selle Italia SLR‘ <dbl>
```

```
# Output: bike_features_tbl_r_tt
```

Step 2 - TRAINING & TEST SETS

- prop: split the data according to a given percentage to train-test sets
- strata: consider different categories when dividing the data set

```
# Fix the random numbers by setting the seed
# This enables the analysis to be reproducible when random numbers are used
set.seed(seed = 1113)
# Put 3/4 of the data into the training set
split_obj <- rsample::initial_split(bike_features_tbl_r_tt, prop   = 0.75,
                                    strata = "category_2")
# Create data frames for the two sets:
train_data <- training(split_obj)
test_data  <- testing(split_obj)
```

Step 3 - Prepare model recipe

```
bikes_data_set <-
                  recipe(price ~ ., data = train_data) %>%
                  step_rm(model_year, weight) %>%
                  update_role(model, category_2, id, new_role = "ID")%>%
                  step_dummy(all_nominal(), -all_outcomes()) %>%
                  prep()

bikes_data_set
```

```
## Data Recipe
##
## Inputs:
##
##       role #variables
##         ID          3
##    outcome          1
##  predictor         32
##
## Training data contained 174 data points and no missing data.
##
## Operations:
##
## Variables removed model_year, weight [trained]
## Dummy variables from model, frame_material, category_2 [trained]
```

```
# Output: bikes_data_set
```

Step 4 - Bake data sets according to the recipe

```
train_transformed_tbl <- bake(bikes_data_set, train_data)
test_transformed_tbl  <- bake(bikes_data_set, test_data)

train_transformed_tbl
```

```
## # A tibble: 174 x 219
##       id ‘SRAM X01 Eagle‘ ‘Shimano Deore ~ ‘Shimano Deore ~ ‘SRAM GX Eagle‘
##    <dbl>            <dbl>            <dbl>            <dbl>           <dbl>
##  1     1                0                0                0               0
##  2     2                0                0                0               0
##  3     3                0                0                0               0
##  4     5                0                0                0               0
##  5     7                0                0                0               0
##  6     8                0                0                0               0
##  7     9                0                0                0               0
##  8    10                0                0                0               0
##  9    11                0                0                0               0
## 10    12                0                0                0               0
## # ... with 164 more rows, and 214 more variables: ‘Shimano 105 R7000 GS‘ <dbl>,
## #   ‘Shimano Ultegra R8000 SGS‘ <dbl>, ‘Shimano Ultegra R8000 SS‘ <dbl>,
## #   ‘Shimano Dura-Ace Di2 R9150, 11-speed‘ <dbl>, ‘Shimano Ultegra Di2 R8050
## #   SS‘ <dbl>, ‘Shimano 105 RS700 2s‘ <dbl>, ‘SRAM X01 Eagle Trigger
## #   12s‘ <dbl>, ‘Shimano Deore XT, 12-speed‘ <dbl>, ‘Shimano Deore M6100
## #   12s‘ <dbl>, ‘Shimano SLX M7100 12s‘ <dbl>, ‘Shimano Deore XTR,
## #   12-speed‘ <dbl>, ‘SRAM NX Eagle Trigger 12s‘ <dbl>, ‘Fizik Essenza‘ <dbl>,
## #   ‘Canyon Sport Saddle EP1249‘ <dbl>, ‘Canyon SD:ON‘ <dbl>, ‘Iridium
## #   Trail‘ <dbl>, ‘Fizik Antares R5‘ <dbl>, ‘Fizik Antares R3‘ <dbl>, ‘Fizik
## #   Mistica‘ <dbl>, ‘Selle Italia Model X‘ <dbl>, ‘Iridium Fitness‘ <dbl>,
## #   ‘Fizik Aliante R5‘ <dbl>, ‘Selle Italia X3 Lady‘ <dbl>, ‘Selle Italia
## #   X3‘ <dbl>, ‘Selle Italia SLR‘ <dbl>, price <int>,
## #   model_Aeroad.CF.SL.8.0 <dbl>, model_Aeroad.CF.SL.8.0.Di2 <dbl>,
## #   model_Aeroad.CF.SL.Disc.8.0 <dbl>, model_Aeroad.CF.SL.Disc.8.0.Di2 <dbl>,
## #   model_Aeroad.CF.SLX.9.0.Di2 <dbl>, model_Aeroad.CF.SLX.Disc.9.0.Di2 <dbl>,
## #   model_Aeroad.CF.SLX.Disc.9.0.ETAP <dbl>, model_Aeroad.WMN.CF.SL.8.0 <dbl>,
## #   model_Commuter.4.0 <dbl>, model_Commuter.5.0 <dbl>,
## #   model_Commuter.6.0 <dbl>, model_Commuter.8.0 <dbl>,
## #   model_Commuter.Sport.8.0 <dbl>, model_Commuter.ON.7 <dbl>,
## #   model_Dude.CF.8.0.Trail <dbl>, model_Dude.CF.9.0.Trail <dbl>,
## #   model_Endurace.AL.6.0 <dbl>, model_Endurace.AL.7.0 <dbl>,
## #   model_Endurace.AL.Disc.6.0 <dbl>, model_Endurace.AL.Disc.7.0 <dbl>,
## #   model_Endurace.CF.7.0 <dbl>, model_Endurace.CF.8.0 <dbl>,
## #   model_Endurace.CF.SL.Disc.7.0 <dbl>, model_Endurace.CF.SL.Disc.8.0 <dbl>,
## #   model_Endurace.CF.SL.Disc.8.0.Aero.Di2 <dbl>,
## #   model_Endurace.CF.SL.Disc.8.0.Di2 <dbl>,
## #   model_Endurace.CF.SL.Disc.8.0.Pro <dbl>,
## #   model_Endurace.CF.SLX.Disc.8.0.ETAP <dbl>,
## #   model_Endurace.CF.SLX.Disc.9.0.Di2 <dbl>,
## #   model_Endurace.CF.SLX.Disc.9.0.ETAP <dbl>, model_Endurace.WMN.AL.7.0 <dbl>,
## #   model_Endurace.WMN.AL.Disc.6.0 <dbl>, model_Endurace.WMN.AL.Disc.7.0 <dbl>,
## #   model_Endurace.WMN.CF.SL.Disc.7.0 <dbl>,
## #   model_Endurace.WMN.CF.SL.Disc.8.0 <dbl>, model_Endurace.ON.7.0 <dbl>,
## #   model_Exceed.CF.5 <dbl>, model_Exceed.CF.5.WMN <dbl>,
## #   model_Exceed.CF.6 <dbl>, model_Exceed.CF.7 <dbl>,
## #   model_Exceed.CF.7.WMN <dbl>, model_Exceed.CF.SLX.9 <dbl>,
## #   model_Exceed.CF.SLX.9.0 <dbl>, model_Exceed.CF.SLX.9.0.Race.LTD <dbl>,
## #   model_Grail.6 <dbl>, model_Grail.7 <dbl>, model_Grail.CF.SL.7 <dbl>,
## #   model_Grail.CF.SL.8 <dbl>, model_Grail.CF.SL.8.Di2 <dbl>,
## #   model_Grail.CF.SL.8.eTap <dbl>, model_Grail.CF.SL.8.WMN.Di2 <dbl>,
## #   model_Grail.CF.SL.8.0.ETAP <dbl>, model_Grail.CF.SLX.8.Di2 <dbl>,
## #   model_Grail.CF.SLX.8.eTap <dbl>, model_Grail.ON.CF.7 <dbl>,
## #   model_Grail.ON.CF.7.WMN <dbl>, model_Grail.ON.CF.8 <dbl>,
## #   model_Grail.ON.CF.8.eTap <dbl>, model_Grand.Canyon.5.WMN <dbl>,
## #   model_Grand.Canyon.6.WMN <dbl>, model_Grand.Canyon.7.WMN <dbl>,
## #   model_Grand.Canyon.8 <dbl>, model_Grand.Canyon.Young.Hero <dbl>,
## #   model_Grand.Canyon.ON.AL.8.0 <dbl>, model_Grand.Canyon.ON.AL.9.0 <dbl>,
## #   model_Grand.Canyon.ON.WMN.AL.8.0 <dbl>, model_Inflite.5 <dbl>,
## #   model_Inflite.CF.SL.6 <dbl>, model_Inflite.CF.SL.8 <dbl>,
## #   model_Inflite.CF.SLX.9 <dbl>, model_Inflite.CF.SLX.9.Team <dbl>,
## #   model_Lux.CF.7.WMN <dbl>, model_Lux.CF.SL.6.0 <dbl>,
## #   model_Lux.CF.SL.8.0 <dbl>, ...
```

```
test_transformed_tbl
```

```
## # A tibble: 57 x 219
##        id ‘SRAM X01 Eagle‘ ‘Shimano Deore ~ ‘Shimano Deore ~ ‘SRAM GX Eagle‘
##     <dbl>            <dbl>            <dbl>            <dbl>           <dbl>
##   1     4                0                0                0               0
##   2     6                0                0                0               0
##   3    19                0                0                0               0
##  4    31                0                0                0               0
##  5    37                0                0                0               0
##  6    42                0                0                0               0
##  7    45                0                0                0               0
##  8    47                0                0                0               0
##  9    64                0                0                0               0
## 10    65                0                0                0               0
## # ... with 47 more rows, and 214 more variables: ‘Shimano 105 R7000 GS‘ <dbl>,
## #   ‘Shimano Ultegra R8000 SGS‘ <dbl>, ‘Shimano Ultegra R8000 SS‘ <dbl>,
## #   ‘Shimano Dura-Ace Di2 R9150, 11-speed‘ <dbl>, ‘Shimano Ultegra Di2 R8050
## #   SS‘ <dbl>, ‘Shimano 105 RS700 2s‘ <dbl>, ‘SRAM X01 Eagle Trigger
## #   12s‘ <dbl>, ‘Shimano Deore XT, 12-speed‘ <dbl>, ‘Shimano Deore M6100
## #   12s‘ <dbl>, ‘Shimano SLX M7100 12s‘ <dbl>, ‘Shimano Deore XTR,
## #   12-speed‘ <dbl>, ‘SRAM NX Eagle Trigger 12s‘ <dbl>, ‘Fizik Essenza‘ <dbl>,
## #   ‘Canyon Sport Saddle EP1249‘ <dbl>, ‘Canyon SD:ON‘ <dbl>, ‘Iridium
## #   Trail‘ <dbl>, ‘Fizik Antares R5‘ <dbl>, ‘Fizik Antares R3‘ <dbl>, ‘Fizik
## #   Mistica‘ <dbl>, ‘Selle Italia Model X‘ <dbl>, ‘Iridium Fitness‘ <dbl>,
## #   ‘Fizik Aliante R5‘ <dbl>, ‘Selle Italia X3 Lady‘ <dbl>, ‘Selle Italia
## #   X3‘ <dbl>, ‘Selle Italia SLR‘ <dbl>, price <int>,
## #   model_Aeroad.CF.SL.8.0 <dbl>, model_Aeroad.CF.SL.8.0.Di2 <dbl>,
## #   model_Aeroad.CF.SL.Disc.8.0 <dbl>, model_Aeroad.CF.SL.Disc.8.0.Di2 <dbl>,
## #   model_Aeroad.CF.SLX.9.0.Di2 <dbl>, model_Aeroad.CF.SLX.Disc.9.0.Di2 <dbl>,
## #   model_Aeroad.CF.SLX.Disc.9.0.ETAP <dbl>, model_Aeroad.WMN.CF.SL.8.0 <dbl>,
## #   model_Commuter.4.0 <dbl>, model_Commuter.5.0 <dbl>,
## #   model_Commuter.6.0 <dbl>, model_Commuter.8.0 <dbl>,
## #   model_Commuter.Sport.8.0 <dbl>, model_Commuter.ON.7 <dbl>,
## #   model_Dude.CF.8.0.Trail <dbl>, model_Dude.CF.9.0.Trail <dbl>,
## #   model_Endurace.AL.6.0 <dbl>, model_Endurace.AL.7.0 <dbl>,
## #   model_Endurace.AL.Disc.6.0 <dbl>, model_Endurace.AL.Disc.7.0 <dbl>,
## #   model_Endurace.CF.7.0 <dbl>, model_Endurace.CF.8.0 <dbl>,
## #   model_Endurace.CF.SL.Disc.7.0 <dbl>, model_Endurace.CF.SL.Disc.8.0 <dbl>,
## #   model_Endurace.CF.SL.Disc.8.0.Aero.Di2 <dbl>,
## #   model_Endurace.CF.SL.Disc.8.0.Di2 <dbl>,
## #   model_Endurace.CF.SL.Disc.8.0.Pro <dbl>,
## #   model_Endurace.CF.SLX.Disc.8.0.ETAP <dbl>,
## #   model_Endurace.CF.SLX.Disc.9.0.Di2 <dbl>,
## #   model_Endurace.CF.SLX.Disc.9.0.ETAP <dbl>, model_Endurace.WMN.AL.7.0 <dbl>,
## #   model_Endurace.WMN.AL.Disc.6.0 <dbl>, model_Endurace.WMN.AL.Disc.7.0 <dbl>,
## #   model_Endurace.WMN.CF.SL.Disc.7.0 <dbl>,
## #   model_Endurace.WMN.CF.SL.Disc.8.0 <dbl>, model_Endurace.ON.7.0 <dbl>,
## #   model_Exceed.CF.5 <dbl>, model_Exceed.CF.5.WMN <dbl>,
## #   model_Exceed.CF.6 <dbl>, model_Exceed.CF.7 <dbl>,
## #   model_Exceed.CF.7.WMN <dbl>, model_Exceed.CF.SLX.9 <dbl>,
## #   model_Exceed.CF.SLX.9.0 <dbl>, model_Exceed.CF.SLX.9.0.Race.LTD <dbl>,
## #   model_Grail.6 <dbl>, model_Grail.7 <dbl>, model_Grail.CF.SL.7 <dbl>,
## #   model_Grail.CF.SL.8 <dbl>, model_Grail.CF.SL.8.Di2 <dbl>,
## #   model_Grail.CF.SL.8.eTap <dbl>, model_Grail.CF.SL.8.WMN.Di2 <dbl>,
## #   model_Grail.CF.SL.8.0.ETAP <dbl>, model_Grail.CF.SLX.8.Di2 <dbl>,
## #   model_Grail.CF.SLX.8.eTap <dbl>, model_Grail.ON.CF.7 <dbl>,
## #   model_Grail.ON.CF.7.WMN <dbl>, model_Grail.ON.CF.8 <dbl>,
## #   model_Grail.ON.CF.8.eTap <dbl>, model_Grand.Canyon.5.WMN <dbl>,
## #   model_Grand.Canyon.6.WMN <dbl>, model_Grand.Canyon.7.WMN <dbl>,
## #   model_Grand.Canyon.8 <dbl>, model_Grand.Canyon.Young.Hero <dbl>,
## #   model_Grand.Canyon.ON.AL.8.0 <dbl>, model_Grand.Canyon.ON.AL.9.0 <dbl>,
## #   model_Grand.Canyon.ON.WMN.AL.8.0 <dbl>, model_Inflite.5 <dbl>,
## #   model_Inflite.CF.SL.6 <dbl>, model_Inflite.CF.SL.8 <dbl>,
## #   model_Inflite.CF.SLX.9 <dbl>, model_Inflite.CF.SLX.9.Team <dbl>,
## #   model_Lux.CF.7.WMN <dbl>, model_Lux.CF.SL.6.0 <dbl>,
## #   model_Lux.CF.SL.8.0 <dbl>, ...
```

Step 5 - Select model and engine to analyze the data

```
lr_mod_b <- linear_reg(mode = "regression") %>%  set_engine("lm")

lr_mod_b
```

```
## Linear Regression Model Specification (regression)
##
## Computational engine: lm
```

Step 6 - Select work flow

```
bikes_workflow <-
  workflow() %>%
  add_model(lr_mod_b) %>%
  add_recipe(bikes_data_set)

bikes_workflow
```

```
## == Workflow ====================================================================
## Preprocessor: Recipe
## Model: linear_reg()
##
## -- Preprocessor ----------------------------------------------------------------
## 2 Recipe Steps
##
## * step_rm()
## * step_dummy()
##
## -- Model -----------------------------------------------------------------------
## Linear Regression Model Specification (regression)
##
## Computational engine: lm
```

Step 7 - Fit the model, to determine model's governing equation

```
# Use purrr to map
bikes_fit <-
  bikes_workflow %>%
  fit(data = train_data)

bikes_fit
```

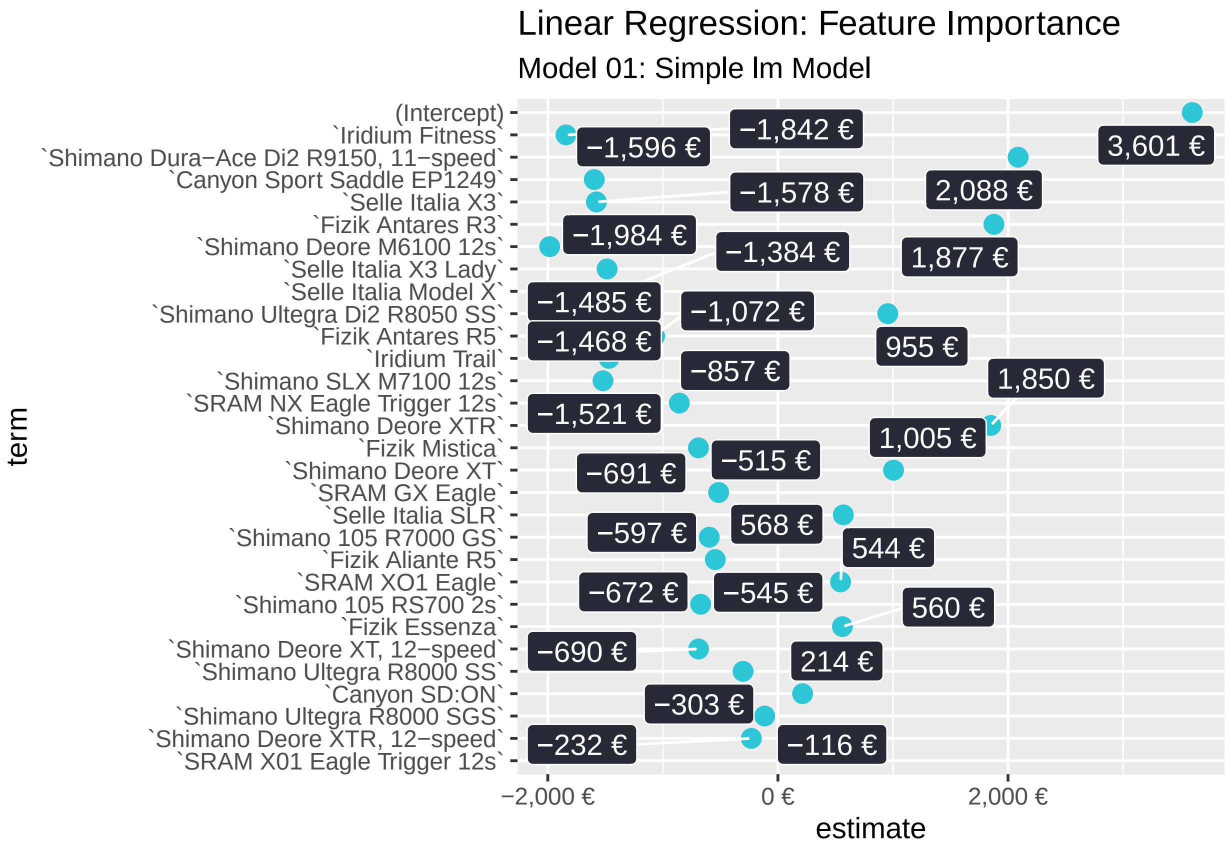
```
## == Workflow [trained] ==========================================================
## Preprocessor: Recipe
## Model: linear_reg()
##
## -- Preprocessor ----------------------------------------------------------------
## 2 Recipe Steps
##
## * step_rm()
## * step_dummy()
##
## -- Model -----------------------------------------------------------------------
##
## Call:
## stats::lm(formula = ..y ~ ., data = data)
##
## Coefficients:
##                            (Intercept)                        `SRAM X01 Eagle`
##                                 3601.2                                   544.4
##                     `Shimano Deore XT`                     `Shimano Deore XTR`
##                                 1005.2                                  1849.7
##                        `SRAM GX Eagle`                  `Shimano 105 R7000 GS`
##                                 -515.1                                  -596.9
##            `Shimano Ultegra R8000 SGS`              `Shimano Ultegra R8000 SS`
##                                 -115.6                                  -303.0
## `Shimano Dura-Ace Di2 R9150, 11-speed`          `Shimano Ultegra Di2 R8050 SS`
##                                 2088.0                                   954.9
##                  `Shimano 105 RS700 2s`            `SRAM X01 Eagle Trigger 12s`
##                                 -671.6                                      NA
##           `Shimano Deore XT, 12-speed`               `Shimano Deore M6100 12s`
##                                 -689.8                                 -1984.0
##                `Shimano SLX M7100 12s`           `Shimano Deore XTR, 12-speed`
##                                -1521.0                                  -231.5
##            `SRAM NX Eagle Trigger 12s`                         `Fizik Essenza`
##                                 -856.9                                   560.3
##           `Canyon Sport Saddle EP1249`                          `Canyon SD:ON`
##                                -1596.3                                   214.4
##                        `Iridium Trail`                      `Fizik Antares R5`
##                                -1468.5                                 -1072.0
##                     `Fizik Antares R3`                         `Fizik Mistica`
##                                 1877.4                                  -690.8
##                 `Selle Italia Model X`                       `Iridium Fitness`
##                                -1384.1                                 -1842.0
##                     `Fizik Aliante R5`                  `Selle Italia X3 Lady`
##                                 -544.6                                 -1485.0
##                      `Selle Italia X3`                      `Selle Italia SLR`
##                                -1577.7                                   568.3
```

```
# Output: bikes_fit
```

```
# Plot the given model

  bikes_fit %>% pull_workflow_fit() %>%
  tidy() %>%
  arrange(p.value) %>%
 mutate(term = as_factor(term) %>% fct_rev()) %>%

  ggplot(aes(x = estimate, y = term)) +
  geom_point(color = "#2dc6d6", size = 3) +
  ggrepel::geom_label_repel(aes(label = scales::dollar(estimate, accuracy = 1, suffix = " €", prefix =
                            size = 4, fill = "#272A36", color = "white") +
  scale_x_continuous(labels = scales::dollar_format(suffix = " €", prefix = "")) +
  labs(title = "Linear Regression: Feature Importance",
       subtitle = "Model 01: Simple lm Model")
```


Linear Regression: Feature Importance

Model 01: Simple lm Model

Step 8 - Prepare for model prediction

```
# Generalized into a function
calc_metrics <- function(model, new_data = test_tbl) {

  model %>%
    predict(new_data = new_data) %>%

    bind_cols(new_data %>% select(price)) %>%
    yardstick::metrics(truth = price, estimate = .pred)

}
```

Step 9 - Evaluate the predicted resutls

```
bikes_fit %>% calc_metrics(train_data)
```

```
## # A tibble: 3 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 rmse    standard    1087.
## 2 rsq     standard       0.594
## 3 mae     standard     796.
```

Test with model: glmnet , with the same steps!

```
#init model:
set.seed(1234)

lm_model_glment <-linear_reg(mode    = "regression",
                                     penalty = 10,
                                     mixture = 0.1) %>%
    set_engine("glmnet")
lm_model_glment
```

```
## Linear Regression Model Specification (regression)
##
## Main Arguments:
##   penalty = 10
##   mixture = 0.1
##
## Computational engine: glmnet
```

```
bikes_workflow_glment <-
  workflow() %>%
  add_model(lm_model_glment) %>%
  add_recipe(bikes_data_set)

bikes_workflow_glment
```

```
## == Workflow ====================================================================
## Preprocessor: Recipe
## Model: linear_reg()
##
## -- Preprocessor ----------------------------------------------------------------
## 2 Recipe Steps
##
## * step_rm()
## * step_dummy()
##
## -- Model -----------------------------------------------------------------------
## Linear Regression Model Specification (regression)
```

```
##
## Main Arguments:
##   penalty = 10
##   mixture = 0.1
##
## Computational engine: glmnet
```

```
bikes_fit_glment <-
  bikes_workflow_glment %>%
  fit(data = train_data)

bikes_fit_glment
```

```
## == Workflow [trained] ==========================================================
## Preprocessor: Recipe
## Model: linear_reg()
##
## -- Preprocessor ----------------------------------------------------------------
## 2 Recipe Steps
##
## * step_rm()
## * step_dummy()
##
## -- Model -----------------------------------------------------------------------
##
## Call:  glmnet::glmnet(x = maybe_matrix(x), y = y, family = "gaussian",      alpha = ~0.1)
##
##    Df  %Dev Lambda
## 1   0  0.00 7625.0
## 2   1  0.75 6948.0
## 3   2  1.89 6330.0
## 4   2  3.16 5768.0
## 5   2  4.43 5256.0
## 6   3  5.79 4789.0
## 7   3  7.43 4363.0
## 8   5  9.30 3976.0
## 9   7 11.61 3622.0
## 10  8 14.12 3301.0
## 11 10 16.91 3007.0
## 12 10 19.65 2740.0
## 13 11 22.35 2497.0
## 14 12 24.99 2275.0
## 15 14 27.66 2073.0
## 16 15 30.32 1889.0
## 17 16 32.80 1721.0
## 18 16 35.11 1568.0
## 19 17 37.24 1429.0
## 20 18 39.28 1302.0
## 21 21 41.27 1186.0
## 22 23 43.14 1081.0
## 23 25 44.90  984.8
## 24 25 46.51  897.3
## 25 25 47.95  817.6
## 26 26 49.25  745.0
## 27 26 50.40  678.8
## 28 26 51.42  618.5
## 29 27 52.32  563.5
## 30 28 53.17  513.5
## 31 28 53.93  467.9
## 32 28 54.59  426.3
## 33 28 55.17  388.4
## 34 28 55.68  353.9
## 35 28 56.13  322.5
## 36 28 56.51  293.8
## 37 28 56.85  267.7
## 38 28 57.14  243.9
## 39 28 57.40  222.3
## 40 29 57.62  202.5
## 41 29 57.84  184.5
## 42 29 58.03  168.1
## 43 29 58.20  153.2
## 44 29 58.35  139.6
## 45 29 58.48  127.2
## 46 29 58.60  115.9
##
## ...
## and 37 more lines.
```

```
bikes_fit_glment %>% calc_metrics(train_data)
```

```
## # A tibble: 3 x 3
##   .metric .estimator .estimate
##   <chr>   <chr>          <dbl>
## 1 rmse    standard    1088.
## 2 rsq     standard       0.594
## 3 mae     standard     795.
```