# Linear

June 10, 2024

[1]:

```r
# It may take several minutes to install those libraries in Watson Studio
install.packages("rlang")
```

```
Updating HTML index of packages in '.Library'
Making 'packages.html' … done
```

[2]:

```r
# It may take several minutes to install those libraries in Watson Studio
library("tidymodels")
library("tidyverse")
library("stringr")
```

```
Warning message:
"replacing previous import ‘lifecycle::last_warnings’ by ‘rlang::last_warnings’
when loading ‘tibble’"Warning message:
"replacing previous import ‘ellipsis::check_dots_unnamed’ by
‘rlang::check_dots_unnamed’ when loading ‘tibble’"Warning message:
"replacing previous import ‘ellipsis::check_dots_used’ by
‘rlang::check_dots_used’ when loading ‘tibble’"Warning message:
"replacing previous import ‘ellipsis::check_dots_empty’ by
‘rlang::check_dots_empty’ when loading ‘tibble’"  Attaching packages
                     tidymodels 0.1.0
 broom     0.5.6       recipes   0.1.12
 dials     0.0.6       rsample   0.0.5
 dplyr     0.8.5       tibble    3.0.1
 ggplot2   3.3.0       tune      0.1.0
 infer     0.5.1       workflows 0.1.1
 parsnip   0.1.0       yardstick 0.0.6
 purrr     0.3.4
  Conflicts                         tidymodels_conflicts()
 purrr::discard()  masks scales::discard()
 dplyr::filter()   masks stats::filter()
 dplyr::lag()      masks stats::lag()
 ggplot2::margin() masks dials::margin()
 recipes::step()   masks stats::step()
  Attaching packages                       tidyverse 1.3.0
 readr   1.3.1      forcats 0.5.0
 stringr 1.4.0
  Conflicts                         tidyverse_conflicts()
 readr::col_factor() masks scales::col_factor()
 purrr::discard()    masks scales::discard()
 dplyr::filter()     masks stats::filter()
 stringr::fixed()    masks recipes::fixed()
 dplyr::lag()        masks stats::lag()
 ggplot2::margin()   masks dials::margin()
 readr::spec()       masks yardstick::spec()
```

[3]:

```
# Dataset URL
dataset_url <- "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.
 ↪cloud/IBMDeveloperSkillsNetwork-RP0321EN-SkillsNetwork/labs/datasets/
 ↪seoul_bike_sharing_converted_normalized.csv"
bike_sharing_df <- read_csv(dataset_url)
spec(bike_sharing_df)
```

```
Parsed with column specification:
cols(
  .default = col_double(),
  DATE = col_character(),
  FUNCTIONING_DAY = col_character()
)
See spec(…) for full column specifications.
```

```
cols(
  DATE = col_character(),
  RENTED_BIKE_COUNT = col_double(),
  TEMPERATURE = col_double(),
  HUMIDITY = col_double(),
  WIND_SPEED = col_double(),
  VISIBILITY = col_double(),
  DEW_POINT_TEMPERATURE = col_double(),
  SOLAR_RADIATION = col_double(),
  RAINFALL = col_double(),
  SNOWFALL = col_double(),
  FUNCTIONING_DAY = col_character(),
  `0` = col_double(),
  `1` = col_double(),
  `10` = col_double(),
  `11` = col_double(),
  `12` = col_double(),
  `13` = col_double(),
  `14` = col_double(),
  `15` = col_double(),
  `16` = col_double(),
  `17` = col_double(),
  `18` = col_double(),
  `19` = col_double(),
  `2` = col_double(),
  `20` = col_double(),
  `21` = col_double(),
  `22` = col_double(),
  `23` = col_double(),
  `3` = col_double(),
  `4` = col_double(),
  `5` = col_double(),
  `6` = col_double(),
  `7` = col_double(),
  `8` = col_double(),
  `9` = col_double(),
  AUTUMN = col_double(),
  SPRING = col_double(),
  SUMMER = col_double(),
  WINTER = col_double(),
  HOLIDAY = col_double(),
  NO_HOLIDAY = col_double()
)
```

[4]:

```
bike_sharing_df <- bike_sharing_df %>%
                   select(-DATE, -FUNCTIONING_DAY)
```

[5]:

```
# Use the `initial_split()`, `training()`, and `testing()` functions to split␣
 ↪the dataset
# With seed 1234
set.seed(1234)
# prop = 3/4
bike_sharing_split <- initial_split(bike_sharing_df, prop = 0.75)
# train_data
train_data <- training(bike_sharing_split)
# test_data
test_data <- testing(bike_sharing_split)
```

[6]:

```
##TASK: Build a linear regression model using weather variables only␣
 ↪lm_model_weather
### Use `linear_reg()` with engine `lm` and mode `regression`
lm_model_weather <- linear_reg( mode ="regression") %>%
# Set engine
set_engine(engine = "lm")
# Print the linear function
lm_model_weather
```

```
Linear Regression Model Specification (regression)

Computational engine: lm
```

[7]:

```
train_fit <- lm_model_weather %>%
fit(RENTED_BIKE_COUNT ~ TEMPERATURE + HUMIDITY + WIND_SPEED + VISIBILITY +␣
 ↪DEW_POINT_TEMPERATURE + SOLAR_RADIATION + RAINFALL + SNOWFALL, data =␣
 ↪train_data)
train_fit
```

```
parsnip model object

Fit time:  7ms

Call:
stats::lm(formula = formula, data = data)

Coefficients:
          (Intercept)            TEMPERATURE               HUMIDITY
              147.647               2452.112               -895.830
           WIND_SPEED             VISIBILITY  DEW_POINT_TEMPERATURE
              402.183                  5.356               -368.982
      SOLAR_RADIATION               RAINFALL               SNOWFALL
             -435.703              -1771.467                354.761
```

[8]:

```
##TASK: Model evaluation and identification of important variables
# Use predict() function to generate test results for `lm_model_weather` and␣
 ↪`lm_model_all`
train_results <- train_fit %>%
# Make the predictions and save the predicted values
predict(new_data = train_data) %>%

# Create a new column to save the true values
mutate(truth = train_data$RENTED_BIKE_COUNT)
head(train_results)
```

A tibble: 6 × 2

| .pred | truth |
|---|---|
| <dbl> | <dbl> |
| 391.5112 | 254 |
| 293.4206 | 204 |
| 250.5649 | 107 |
| 309.0461 | 100 |
| 228.2566 | 460 |
| 241.6479 | 930 |

[9]:

```
test_results <- train_fit %>%
# Make the predictions and save the predicted values
predict(new_data = test_data) %>%
# Create a new column to save the true values
mutate(truth = test_data$RENTED_BIKE_COUNT)
```

```r
head(test_results)
```

A tibble: 6 × 2

| .pred<br>&lt;dbl&gt; | truth<br>&lt;dbl&gt; |
|---|---|
| 274.3531 | 173 |
| 378.1757 | 78 |
| 312.9917 | 181 |
| 336.7766 | 490 |
| 631.8560 | 449 |
| 638.0489 | 451 |

[10]:
```r
lm_model_all <- linear_reg( mode ="regression") %>%
# Set engine
set_engine(engine = "lm")
# Print the linear function
lm_model_all
```

```
Linear Regression Model Specification (regression)

Computational engine: lm
```

[11]:
```r
train_fit2 <- lm_model_all %>%
fit(RENTED_BIKE_COUNT ~ ., data = train_data)
train_fit2
```

```
parsnip model object

Fit time:  13ms

Call:
stats::lm(formula = formula, data = data)

Coefficients:
          (Intercept)            TEMPERATURE               HUMIDITY
              216.584                810.604               -920.587
           WIND_SPEED             VISIBILITY  DEW_POINT_TEMPERATURE
               -9.313                 24.368                632.384
      SOLAR_RADIATION               RAINFALL               SNOWFALL
              249.752              -1982.940                232.451
                  `0`                    `1`                   `10`
              -12.500               -125.701               -222.233
                 `11`                   `12`                   `13`
             -231.428               -194.605               -189.528
                 `14`                   `15`                   `16`
             -183.060               -103.127                 43.598
                 `17`                   `18`                   `19`
              306.732                782.842                519.456
```

```
                `2`                  `20`                  `21`
           -259.946               394.635               430.185
               `22`                  `23`                   `3`
            322.664                90.181              -327.245
                `4`                   `5`                   `6`
           -389.558              -379.055              -216.689
                `7`                   `8`                   `9`
            110.492               502.354                    NA
             AUTUMN                SPRING                SUMMER
            357.978               194.421               172.901
             WINTER               HOLIDAY            NO_HOLIDAY
                 NA              -129.167                    NA
```

[12]:

```r
train_results2 <- train_fit2 %>%
# Make the predictions and save the predicted values
predict(new_data = train_data) %>%

# Create a new column to save the true values
mutate(truth = train_data$RENTED_BIKE_COUNT)
head(train_results2)
```

```
Warning message in predict.lm(object = object$fit, newdata = new_data, type =
"response"):
"prediction from a rank-deficient fit may be misleading"
```

A tibble: 6 × 2

| .pred<br>&lt;dbl&gt; | truth<br>&lt;dbl&gt; |
|---|---|
| 198.90633 | 254 |
| 73.82145 | 204 |
| -156.55532 | 107 |
| -195.80875 | 100 |
| 264.36386 | 460 |
| 657.77254 | 930 |

[13]:

```r
test_results2 <- train_fit2 %>%
# Make the predictions and save the predicted values
predict(new_data = test_data) %>%
# Create a new column to save the true values
mutate(truth = test_data$RENTED_BIKE_COUNT)
head(test_results2)
```

```
Warning message in predict.lm(object = object$fit, newdata = new_data, type =
"response"):
"prediction from a rank-deficient fit may be misleading"
```

A tibble: 6 × 2

| .pred<br>&lt;dbl&gt; | truth<br>&lt;dbl&gt; |
|---|---|
| -78.24847 | 173 |
| -191.16239 | 78 |
| -25.99046 | 181 |
| 251.97376 | 490 |
| 330.23653 | 449 |
| 347.24700 | 451 |

[14]:

```
rsq(test_results, truth = truth,
estimate = .pred)

rsq(test_results2, truth = truth,
estimate = .pred)
```

A tibble: 1 × 3

| .metric<br>&lt;chr&gt; | .estimator<br>&lt;chr&gt; | .estimate<br>&lt;dbl&gt; |
|---|---|---|
| rsq | standard | 0.4458692 |

A tibble: 1 × 3

| .metric<br>&lt;chr&gt; | .estimator<br>&lt;chr&gt; | .estimate<br>&lt;dbl&gt; |
|---|---|---|
| rsq | standard | 0.6592615 |

[15]:

```
rmse(test_results, truth = truth,
estimate = .pred)

rmse(test_results2, truth = truth,
estimate = .pred)
```

A tibble: 1 × 3

| .metric<br>&lt;chr&gt; | .estimator<br>&lt;chr&gt; | .estimate<br>&lt;dbl&gt; |
|---|---|---|
| rmse | standard | 470.4436 |

A tibble: 1 × 3

| .metric<br>&lt;chr&gt; | .estimator<br>&lt;chr&gt; | .estimate<br>&lt;dbl&gt; |
|---|---|---|
| rmse | standard | 369.1709 |

[16]:

```
## summary(lm_model_all$fit)
lm_model_all$coefficients
```

```
NULL
```

[17]:

```
lm_model_weather$coefficients
```

```
NULL
```

[18]:

```
train_fit2  %>%
  tidy() %>%
  arrange(desc(abs(estimate)))
```

A tibble: 36 × 5

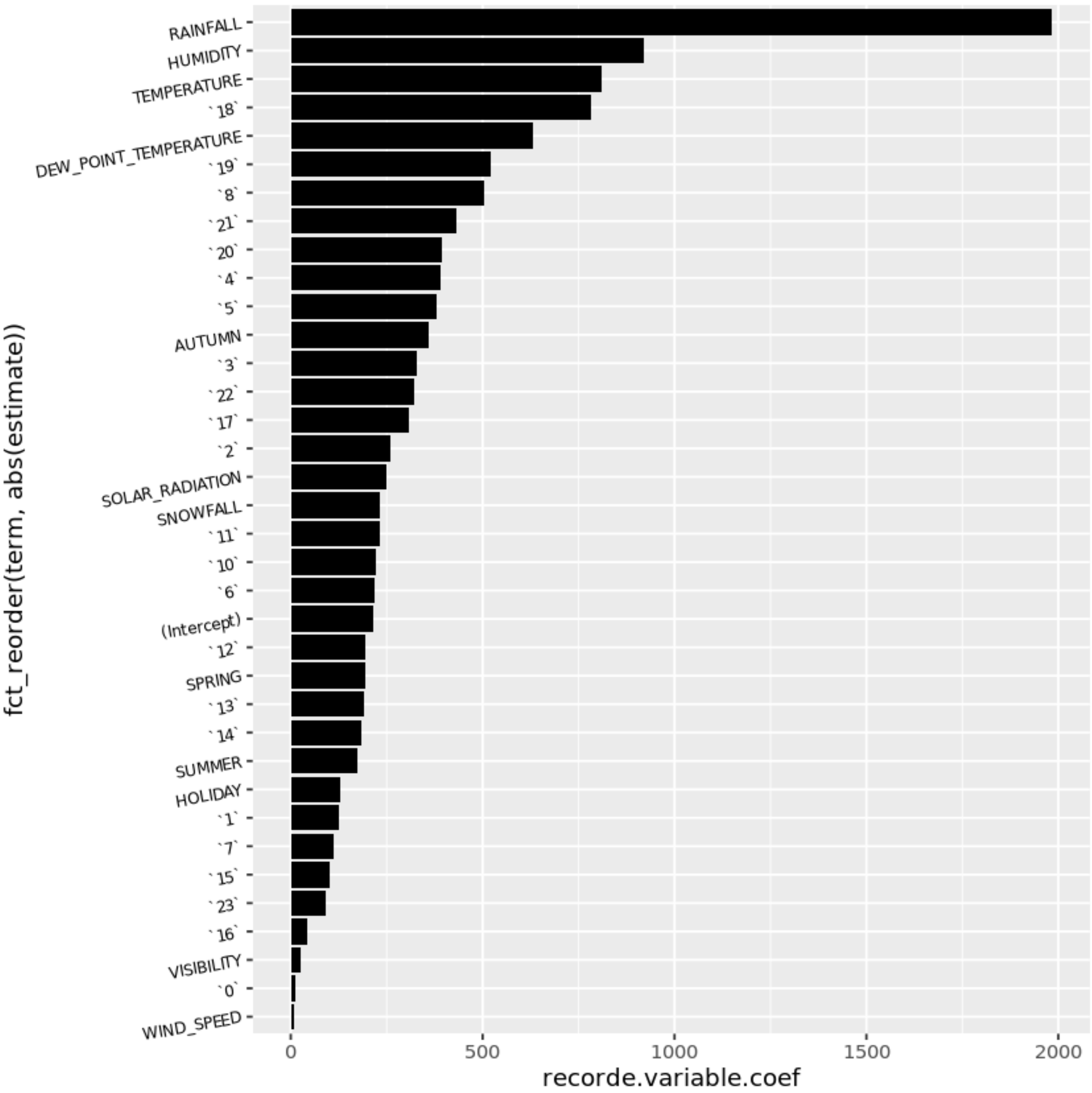
| term<br><chr> | estimate<br><dbl> | std.error<br><dbl> | statistic<br><dbl> | p.value<br><dbl> |
|---|---|---|---|---|
| RAINFALL | -1982.940423 | 143.64226 | -13.8047146 | 9.922612e-43 |
| HUMIDITY | -920.586909 | 97.24275 | -9.4668947 | 3.978466e-21 |
| TEMPERATURE | 810.603987 | 209.03372 | 3.8778623 | 1.064486e-04 |
| ‘18‘ | 782.841507 | 34.51207 | 22.6831196 | 1.428163e-109 |
| DEW_POINT_TEMPERATURE | 632.384298 | 217.47791 | 2.9078093 | 3.652385e-03 |
| ‘19‘ | 519.455508 | 34.54060 | 15.0389836 | 3.001427e-50 |
| ‘8‘ | 502.353808 | 33.50621 | 14.9928559 | 5.875496e-50 |
| ‘21‘ | 430.185215 | 34.33684 | 12.5283880 | 1.380106e-35 |
| ‘20‘ | 394.634676 | 34.72886 | 11.3633072 | 1.234052e-29 |
| ‘4‘ | -389.558392 | 34.03286 | -11.4465363 | 4.834550e-30 |
| ‘5‘ | -379.055209 | 33.81771 | -11.2087790 | 6.910457e-29 |
| AUTUMN | 357.978265 | 19.99659 | 17.9019693 | 5.967468e-70 |
| ‘3‘ | -327.244955 | 33.88966 | -9.6561893 | 6.555260e-22 |
| ‘22‘ | 322.663949 | 33.93497 | 9.5083031 | 2.689464e-21 |
| ‘17‘ | 306.732435 | 34.27155 | 8.9500589 | 4.599800e-19 |
| ‘2‘ | -259.946174 | 34.57530 | -7.5182614 | 6.322812e-14 |
| SOLAR_RADIATION | 249.752454 | 41.34675 | 6.0404379 | 1.624462e-09 |
| SNOWFALL | 232.450790 | 104.60901 | 2.2220915 | 2.631243e-02 |
| ‘11‘ | -231.428411 | 33.98053 | -6.8106172 | 1.061643e-11 |
| ‘10‘ | -222.233262 | 33.31558 | -6.6705509 | 2.764896e-11 |
| ‘6‘ | -216.689105 | 34.16556 | -6.3423250 | 2.419940e-10 |
| (Intercept) | 216.584059 | 50.67322 | 4.2741329 | 1.947061e-05 |
| ‘12‘ | -194.605041 | 34.55857 | -5.6311662 | 1.867030e-08 |
| SPRING | 194.421437 | 19.21791 | 10.1166801 | 7.080230e-24 |
| ‘13‘ | -189.528426 | 35.19884 | -5.3845072 | 7.526455e-08 |
| ‘14‘ | -183.060291 | 35.24584 | -5.1938130 | 2.124823e-07 |
| SUMMER | 172.901302 | 29.04270 | 5.9533473 | 2.768259e-09 |
| HOLIDAY | -129.167078 | 22.44739 | -5.7542142 | 9.112956e-09 |
| ‘1‘ | -125.701499 | 34.57857 | -3.6352430 | 2.799291e-04 |
| ‘7‘ | 110.491959 | 33.93040 | 3.2564298 | 1.134185e-03 |
| ‘15‘ | -103.126885 | 35.09640 | -2.9383890 | 3.311170e-03 |
| ‘23‘ | 90.180803 | 33.97406 | 2.6544016 | 7.964847e-03 |
| ‘16‘ | 43.597895 | 34.23026 | 1.2736652 | 2.028290e-01 |
| VISIBILITY | 24.368049 | 20.23481 | 1.2042639 | 2.285328e-01 |
| ‘0‘ | -12.499804 | 34.01006 | -0.3675326 | 7.132341e-01 |
| WIND_SPEED | -9.313319 | 40.26779 | -0.2312846 | 8.171012e-01 |

[19]:

```
train_fit2 %>%
  tidy() %>%
  filter(!is.na(estimate)) %>%
  ggplot(aes(x = fct_reorder(term, abs(estimate)), y = abs(estimate))) +
  geom_bar(stat = "identity", fill = "black") +
  coord_flip() +
  theme(axis.text.y = element_text(angle = 10, colour = "black", size = 7)) +
  ylab("recorde.variable.coef")
  xlab("Coef")
```

```
$x
[1] "Coef"

attr(,"class")
[1] "labels"
```



[20]:

```
# Dataset URL
dataset_url <- "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.
↪cloud/IBMDeveloperSkillsNetwork-RP0321EN-SkillsNetwork/labs/datasets/
↪seoul_bike_sharing_converted.csv"
bike_sharing_df <- read_csv(dataset_url)
```

```r
spec(bike_sharing_df)
```

```
Parsed with column specification:
cols(
  .default = col_double(),
  DATE = col_character(),
  FUNCTIONING_DAY = col_character()
)
See spec(…) for full column specifications.
```

```
cols(
  DATE = col_character(),
  RENTED_BIKE_COUNT = col_double(),
  TEMPERATURE = col_double(),
  HUMIDITY = col_double(),
  WIND_SPEED = col_double(),
  VISIBILITY = col_double(),
  DEW_POINT_TEMPERATURE = col_double(),
  SOLAR_RADIATION = col_double(),
  RAINFALL = col_double(),
  SNOWFALL = col_double(),
  FUNCTIONING_DAY = col_character(),
  `0` = col_double(),
  `1` = col_double(),
  `10` = col_double(),
  `11` = col_double(),
  `12` = col_double(),
  `13` = col_double(),
  `14` = col_double(),
  `15` = col_double(),
  `16` = col_double(),
  `17` = col_double(),
  `18` = col_double(),
  `19` = col_double(),
  `2` = col_double(),
  `20` = col_double(),
  `21` = col_double(),
  `22` = col_double(),
  `23` = col_double(),
  `3` = col_double(),
  `4` = col_double(),
  `5` = col_double(),
  `6` = col_double(),
  `7` = col_double(),
  `8` = col_double(),
  `9` = col_double(),
  AUTUMN = col_double(),
  SPRING = col_double(),
  SUMMER = col_double(),
  WINTER = col_double(),
  HOLIDAY = col_double(),
  NO_HOLIDAY = col_double()
)
```

[21]:

```
set.seed(1234)
# prop = 3/4
bike_sharing_split2 <- initial_split(bike_sharing_df, prop = 0.75)
# train_data
train_data1 <- training(bike_sharing_split2)
# test_data
test_data2 <- testing(bike_sharing_split2)
```

[22]:

```
lm_model_weather <- linear_reg( mode ="regression") %>%
# Set engine
set_engine(engine = "lm")
# Print the linear function
lm_model_weather
```

```
Linear Regression Model Specification (regression)

Computational engine: lm
```

[23]:

```
train_fit <- lm_model_weather %>%
fit(RENTED_BIKE_COUNT ~ TEMPERATURE + HUMIDITY + WIND_SPEED + VISIBILITY +␣
 ↪DEW_POINT_TEMPERATURE + SOLAR_RADIATION + RAINFALL + SNOWFALL, data =␣
 ↪train_data)
train_fit
```

```
parsnip model object

Fit time:  4ms

Call:
stats::lm(formula = formula, data = data)

Coefficients:
          (Intercept)            TEMPERATURE               HUMIDITY
              147.647               2452.112               -895.830
           WIND_SPEED             VISIBILITY  DEW_POINT_TEMPERATURE
              402.183                  5.356               -368.982
      SOLAR_RADIATION               RAINFALL               SNOWFALL
             -435.703              -1771.467                354.761
```

[24]:

```
test_results <- train_fit %>%
# Make the predictions and save the predicted values
predict(new_data = test_data) %>%
# Create a new column to save the true values
mutate(truth = test_data$RENTED_BIKE_COUNT)
head(test_results)
```

A tibble: 6 × 2

| .pred<br>&lt;dbl&gt; | truth<br>&lt;dbl&gt; |
|---|---|
| 274.3531 | 173 |
| 378.1757 | 78 |
| 312.9917 | 181 |
| 336.7766 | 490 |
| 631.8560 | 449 |
| 638.0489 | 451 |

[25]:

```
lm_model_all <- linear_reg( mode ="regression") %>%
# Set engine
set_engine(engine = "lm")
# Print the linear function
lm_model_all
```

```
Linear Regression Model Specification (regression)

Computational engine: lm
```

[26]:

```
train_fit2 <- lm_model_all %>%
fit(RENTED_BIKE_COUNT ~ ., data = train_data)
train_fit2
```

```
parsnip model object

Fit time:  18ms

Call:
stats::lm(formula = formula, data = data)

Coefficients:
          (Intercept)            TEMPERATURE               HUMIDITY
              216.584                810.604               -920.587
           WIND_SPEED             VISIBILITY  DEW_POINT_TEMPERATURE
               -9.313                 24.368                632.384
      SOLAR_RADIATION               RAINFALL               SNOWFALL
              249.752              -1982.940                232.451
                  `0`                    `1`                   `10`
              -12.500               -125.701               -222.233
                 `11`                   `12`                   `13`
```

```
   -231.428      -194.605      -189.528
       `14`          `15`          `16`
   -183.060      -103.127        43.598
       `17`          `18`          `19`
    306.732       782.842       519.456
        `2`          `20`          `21`
   -259.946       394.635       430.185
       `22`          `23`           `3`
    322.664        90.181      -327.245
        `4`           `5`           `6`
   -389.558      -379.055      -216.689
        `7`           `8`           `9`
    110.492       502.354            NA
     AUTUMN        SPRING        SUMMER
    357.978       194.421       172.901
     WINTER       HOLIDAY    NO_HOLIDAY
         NA      -129.167            NA
```

[27]:
```r
test_results2 <- train_fit2 %>%
# Make the predictions and save the predicted values
predict(new_data = test_data) %>%
# Create a new column to save the true values
mutate(truth = test_data$RENTED_BIKE_COUNT)
head(test_results2)
```

```
Warning message in predict.lm(object = object$fit, newdata = new_data, type =
"response"):
"prediction from a rank-deficient fit may be misleading"
```

A tibble: 6 × 2

| .pred<br>&lt;dbl&gt; | truth<br>&lt;dbl&gt; |
|---|---|
| -78.24847 | 173 |
| -191.16239 | 78 |
| -25.99046 | 181 |
| 251.97376 | 490 |
| 330.23653 | 449 |
| 347.24700 | 451 |

[28]:
```r
rsq(test_results, truth = truth,
estimate = .pred)
rsq(test_results2, truth = truth,
estimate = .pred)

rmse(test_results, truth = truth,
estimate = .pred)
rmse(test_results2, truth = truth,
estimate = .pred)
```

A tibble: 1 × 3

| .metric<br>&lt;chr&gt; | .estimator<br>&lt;chr&gt; | .estimate<br>&lt;dbl&gt; |
|---|---|---|
| rsq | standard | 0.4458692 |

A tibble: 1 × 3

| .metric<br>&lt;chr&gt; | .estimator<br>&lt;chr&gt; | .estimate<br>&lt;dbl&gt; |
|---|---|---|
| rsq | standard | 0.6592615 |

A tibble: 1 × 3

| .metric<br>&lt;chr&gt; | .estimator<br>&lt;chr&gt; | .estimate<br>&lt;dbl&gt; |
|---|---|---|
| rmse | standard | 470.4436 |

A tibble: 1 × 3

| .metric<br>&lt;chr&gt; | .estimator<br>&lt;chr&gt; | .estimate<br>&lt;dbl&gt; |
|---|---|---|
| rmse | standard | 369.1709 |

[29]:

```
lm_model_all$fit$coefficients
```

```
NULL
```

[30]:

```
train_fit2  %>%
  tidy() %>%
  arrange(desc(abs(estimate)))
```

A tibble: 36 × 5

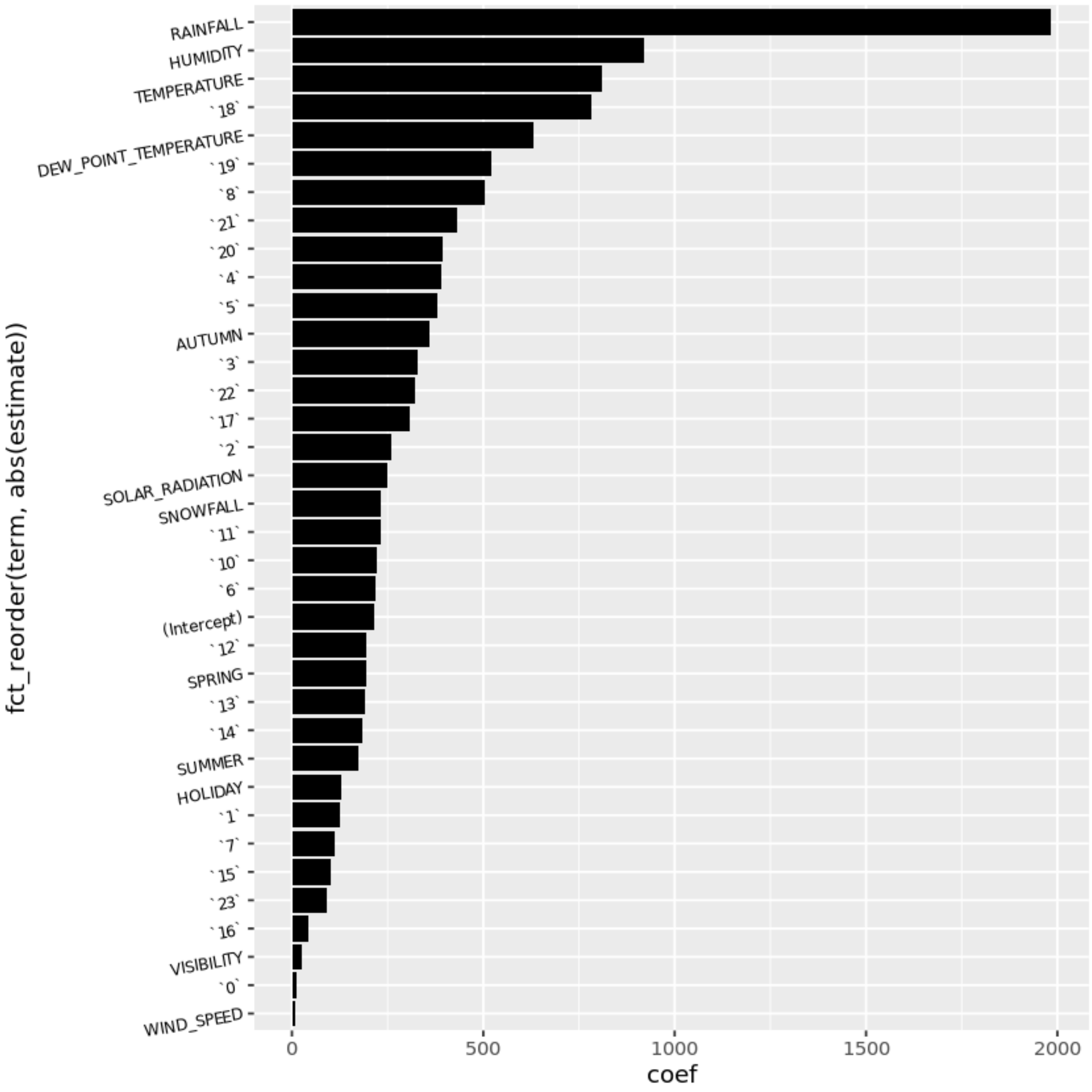
| term | estimate | std.error | statistic | p.value |
|---|---|---|---|---|
| <chr> | <dbl> | <dbl> | <dbl> | <dbl> |
| RAINFALL | -1982.940423 | 143.64226 | -13.8047146 | 9.922612e-43 |
| HUMIDITY | -920.586909 | 97.24275 | -9.4668947 | 3.978466e-21 |
| TEMPERATURE | 810.603987 | 209.03372 | 3.8778623 | 1.064486e-04 |
| ‘18‘ | 782.841507 | 34.51207 | 22.6831196 | 1.428163e-109 |
| DEW_POINT_TEMPERATURE | 632.384298 | 217.47791 | 2.9078093 | 3.652385e-03 |
| ‘19‘ | 519.455508 | 34.54060 | 15.0389836 | 3.001427e-50 |
| ‘8‘ | 502.353808 | 33.50621 | 14.9928559 | 5.875496e-50 |
| ‘21‘ | 430.185215 | 34.33684 | 12.5283880 | 1.380106e-35 |
| ‘20‘ | 394.634676 | 34.72886 | 11.3633072 | 1.234052e-29 |
| ‘4‘ | -389.558392 | 34.03286 | -11.4465363 | 4.834550e-30 |
| ‘5‘ | -379.055209 | 33.81771 | -11.2087790 | 6.910457e-29 |
| AUTUMN | 357.978265 | 19.99659 | 17.9019693 | 5.967468e-70 |
| ‘3‘ | -327.244955 | 33.88966 | -9.6561893 | 6.555260e-22 |
| ‘22‘ | 322.663949 | 33.93497 | 9.5083031 | 2.689464e-21 |
| ‘17‘ | 306.732435 | 34.27155 | 8.9500589 | 4.599800e-19 |
| ‘2‘ | -259.946174 | 34.57530 | -7.5182614 | 6.322812e-14 |
| SOLAR_RADIATION | 249.752454 | 41.34675 | 6.0404379 | 1.624462e-09 |
| SNOWFALL | 232.450790 | 104.60901 | 2.2220915 | 2.631243e-02 |
| ‘11‘ | -231.428411 | 33.98053 | -6.8106172 | 1.061643e-11 |
| ‘10‘ | -222.233262 | 33.31558 | -6.6705509 | 2.764896e-11 |
| ‘6‘ | -216.689105 | 34.16556 | -6.3423250 | 2.419940e-10 |
| (Intercept) | 216.584059 | 50.67322 | 4.2741329 | 1.947061e-05 |
| ‘12‘ | -194.605041 | 34.55857 | -5.6311662 | 1.867030e-08 |
| SPRING | 194.421437 | 19.21791 | 10.1166801 | 7.080230e-24 |
| ‘13‘ | -189.528426 | 35.19884 | -5.3845072 | 7.526455e-08 |
| ‘14‘ | -183.060291 | 35.24584 | -5.1938130 | 2.124823e-07 |
| SUMMER | 172.901302 | 29.04270 | 5.9533473 | 2.768259e-09 |
| HOLIDAY | -129.167078 | 22.44739 | -5.7542142 | 9.112956e-09 |
| ‘1‘ | -125.701499 | 34.57857 | -3.6352430 | 2.799291e-04 |
| ‘7‘ | 110.491959 | 33.93040 | 3.2564298 | 1.134185e-03 |
| ‘15‘ | -103.126885 | 35.09640 | -2.9383890 | 3.311170e-03 |
| ‘23‘ | 90.180803 | 33.97406 | 2.6544016 | 7.964847e-03 |
| ‘16‘ | 43.597895 | 34.23026 | 1.2736652 | 2.028290e-01 |
| VISIBILITY | 24.368049 | 20.23481 | 1.2042639 | 2.285328e-01 |
| ‘0‘ | -12.499804 | 34.01006 | -0.3675326 | 7.132341e-01 |
| WIND_SPEED | -9.313319 | 40.26779 | -0.2312846 | 8.171012e-01 |

[31]:

```
train_fit2 %>%
  tidy() %>%
  filter(!is.na(estimate)) %>%
  ggplot(aes(x = fct_reorder(term, abs(estimate)), y = abs(estimate))) +
  geom_bar(stat = "identity", fill = "black") +
  coord_flip() +
  theme(axis.text.y = element_text(angle = 10, colour = "black", size = 7)) +
  ylab("coef")
  xlab("recorde.variable.coef")
```

```
$x
[1] "recorde.variable.coef"

attr(,"class")
[1] "labels"
```



[ ]:

```

```