Module3\_Assignment1

MTracka

5/31/2021

Load libraries

library(tidyverse)

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

## v ggplot2 3.3.3 v purrr 0.3.4  
## v tibble 3.1.2 v dplyr 1.0.6  
## v tidyr 1.1.3 v stringr 1.4.0  
## v readr 1.4.0 v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(lubridate)

##   
## Attaching package: 'lubridate'

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

library(tidymodels)

## -- Attaching packages -------------------------------------- tidymodels 0.1.3 --

## v broom 0.7.6 v rsample 0.1.0   
## v dials 0.0.9 v tune 0.1.5   
## v infer 0.5.4 v workflows 0.2.2   
## v modeldata 0.1.0 v workflowsets 0.0.2   
## v parsnip 0.1.5 v yardstick 0.0.8   
## v recipes 0.1.16

## -- Conflicts ----------------------------------------- tidymodels\_conflicts() --  
## x scales::discard() masks purrr::discard()  
## x dplyr::filter() masks stats::filter()  
## x recipes::fixed() masks stringr::fixed()  
## x dplyr::lag() masks stats::lag()  
## x yardstick::spec() masks readr::spec()  
## x recipes::step() masks stats::step()  
## \* Use tidymodels\_prefer() to resolve common conflicts.

Read in the dataset “bike\_cleaned-2.csv”

bike = read\_csv("bike\_cleaned-2.csv")

##   
## -- Column specification --------------------------------------------------------  
## cols(  
## instant = col\_double(),  
## dteday = col\_character(),  
## season = col\_character(),  
## mnth = col\_character(),  
## hr = col\_double(),  
## holiday = col\_character(),  
## weekday = col\_character(),  
## workingday = col\_character(),  
## weathersit = col\_character(),  
## temp = col\_double(),  
## atemp = col\_double(),  
## hum = col\_double(),  
## windspeed = col\_double(),  
## casual = col\_double(),  
## registered = col\_double(),  
## count = col\_double()  
## )

Convert “dteday” variable from character to Date

bike = bike %>% mutate(dteday = mdy(dteday))

Convert remaining character variables to factors

bike = bike %>% mutate(season = as\_factor(season))  
bike = bike %>% mutate(mnth = as\_factor(mnth))  
bike = bike %>% mutate(holiday = as\_factor(holiday))  
bike = bike %>% mutate(weekday = as\_factor(weekday))  
bike = bike %>% mutate(workingday = as\_factor(workingday))  
bike = bike %>% mutate(weathersit = as\_factor(weathersit))

Finally convert “hr” variable from numerical to factor.

bike = bike %>% mutate(hr = as\_factor(hr))

###### Task 1.

# Split the data into training and testing sets.

# The training set has 70% of the data, used random seed of 1234,as strata “count”

set.seed(1234)  
bike\_split = initial\_split(bike, prop = 0.70, strata = count)  
train = training(bike\_split)  
test = testing(bike\_split)

###### Task 2.

# Number of rows for train and test data sets.

str(train)

## tibble [12,163 x 16] (S3: tbl\_df/tbl/data.frame)  
## $ instant : num [1:12163] 2 3 4 5 6 7 9 10 11 19 ...  
## $ dteday : Date[1:12163], format: "2011-01-01" "2011-01-01" ...  
## $ season : Factor w/ 4 levels "Winter","Spring",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ mnth : Factor w/ 12 levels "Jan","Feb","Mar",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ hr : Factor w/ 24 levels "0","1","2","3",..: 2 3 4 5 6 7 9 10 11 19 ...  
## $ holiday : Factor w/ 2 levels "NotHoliday","Holiday": 1 1 1 1 1 1 1 1 1 1 ...  
## $ weekday : Factor w/ 7 levels "Saturday","Sunday",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ workingday: Factor w/ 2 levels "NotWorkingDay",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ weathersit: Factor w/ 4 levels "NoPrecip","Misty",..: 1 1 1 1 2 1 1 1 1 3 ...  
## $ temp : num [1:12163] 0.22 0.22 0.24 0.24 0.24 0.22 0.24 0.32 0.38 0.42 ...  
## $ atemp : num [1:12163] 0.273 0.273 0.288 0.288 0.258 ...  
## $ hum : num [1:12163] 0.8 0.8 0.75 0.75 0.75 0.8 0.75 0.76 0.76 0.88 ...  
## $ windspeed : num [1:12163] 0 0 0 0 0.0896 ...  
## $ casual : num [1:12163] 8 5 3 0 0 2 1 8 12 9 ...  
## $ registered: num [1:12163] 32 27 10 1 1 0 7 6 24 26 ...  
## $ count : num [1:12163] 40 32 13 1 1 2 8 14 36 35 ...

# There are 12163 rows in “train” data set.

str(test)

## tibble [5,216 x 16] (S3: tbl\_df/tbl/data.frame)  
## $ instant : num [1:5216] 1 8 21 25 27 36 37 38 39 49 ...  
## $ dteday : Date[1:5216], format: "2011-01-01" "2011-01-01" ...  
## $ season : Factor w/ 4 levels "Winter","Spring",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ mnth : Factor w/ 12 levels "Jan","Feb","Mar",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ hr : Factor w/ 24 levels "0","1","2","3",..: 1 8 21 1 3 13 14 15 16 2 ...  
## $ holiday : Factor w/ 2 levels "NotHoliday","Holiday": 1 1 1 1 1 1 1 1 1 1 ...  
## $ weekday : Factor w/ 7 levels "Saturday","Sunday",..: 1 1 1 2 2 2 2 2 2 3 ...  
## $ workingday: Factor w/ 2 levels "NotWorkingDay",..: 1 1 1 1 1 1 1 1 1 2 ...  
## $ weathersit: Factor w/ 4 levels "NoPrecip","Misty",..: 1 1 2 2 2 2 2 3 3 1 ...  
## $ temp : num [1:5216] 0.24 0.2 0.4 0.46 0.42 0.36 0.36 0.36 0.34 0.2 ...  
## $ atemp : num [1:5216] 0.288 0.258 0.409 0.455 0.424 ...  
## $ hum : num [1:5216] 0.81 0.86 0.87 0.88 1 0.66 0.66 0.76 0.81 0.44 ...  
## $ windspeed : num [1:5216] 0 0 0.254 0.298 0.284 ...  
## $ casual : num [1:5216] 3 1 11 4 1 20 11 4 19 0 ...  
## $ registered: num [1:5216] 13 2 25 13 8 73 64 55 55 2 ...  
## $ count : num [1:5216] 16 3 36 17 9 93 75 59 74 2 ...

# There are 5216 rows in “test” data set.

###### Task 3.

# Build a linear regression model using training set and predict “count” using variables “season”, “mnth”, “hr”, “holiday”, “weekday”, “temp”, “weathersit”.

# Create recipe.

train\_recipe = recipe(count ~ season + mnth + hr + holiday + weekday + temp + weathersit , train)  
lm\_model =  
linear\_reg() %>%  
set\_engine("lm")

# Create lm\_wflow

lm\_wflow =  
workflow() %>%  
add\_model(lm\_model) %>%  
add\_recipe(train\_recipe)

lm\_fit = fit(lm\_wflow, train)

summary(lm\_fit$fit$fit$fit)

##   
## Call:  
## stats::lm(formula = ..y ~ ., data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -427.33 -62.08 -9.82 51.84 503.54   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -81.6699 6.9466 -11.757 < 2e-16 \*\*\*  
## seasonSpring 27.4972 6.3951 4.300 1.72e-05 \*\*\*  
## seasonSummer 18.7645 7.5881 2.473 0.01342 \*   
## seasonFall 62.5367 6.4533 9.691 < 2e-16 \*\*\*  
## mnthFeb -0.5997 5.1373 -0.117 0.90707   
## mnthMar 3.0778 5.7904 0.532 0.59506   
## mnthApr -1.3130 8.6231 -0.152 0.87898   
## mnthMay -2.6894 9.2230 -0.292 0.77060   
## mnthJun -15.8125 9.4879 -1.667 0.09562 .   
## mnthJul -40.2300 10.6077 -3.793 0.00015 \*\*\*  
## mnthAug -16.4993 10.3574 -1.593 0.11119   
## mnthSep 3.9859 9.2187 0.432 0.66548   
## mnthOct -3.0817 8.5334 -0.361 0.71800   
## mnthNov -14.7632 8.2403 -1.792 0.07322 .   
## mnthDec -16.2734 6.5606 -2.480 0.01313 \*   
## hr1 -20.7836 6.9908 -2.973 0.00295 \*\*   
## hr2 -29.0673 6.9980 -4.154 3.29e-05 \*\*\*  
## hr3 -41.4592 7.0968 -5.842 5.29e-09 \*\*\*  
## hr4 -41.2506 7.0386 -5.861 4.73e-09 \*\*\*  
## hr5 -27.2665 6.9794 -3.907 9.41e-05 \*\*\*  
## hr6 31.8318 7.0125 4.539 5.70e-06 \*\*\*  
## hr7 164.5446 7.0278 23.413 < 2e-16 \*\*\*  
## hr8 305.3583 6.9782 43.759 < 2e-16 \*\*\*  
## hr9 163.9524 7.0096 23.390 < 2e-16 \*\*\*  
## hr10 105.9395 6.9986 15.137 < 2e-16 \*\*\*  
## hr11 138.1987 6.9861 19.782 < 2e-16 \*\*\*  
## hr12 179.5246 6.9799 25.720 < 2e-16 \*\*\*  
## hr13 177.5739 7.0533 25.176 < 2e-16 \*\*\*  
## hr14 152.0364 7.1106 21.382 < 2e-16 \*\*\*  
## hr15 170.3496 7.0967 24.004 < 2e-16 \*\*\*  
## hr16 229.1493 7.1110 32.225 < 2e-16 \*\*\*  
## hr17 384.6252 7.0221 54.774 < 2e-16 \*\*\*  
## hr18 342.3854 7.0387 48.643 < 2e-16 \*\*\*  
## hr19 236.7980 7.0437 33.618 < 2e-16 \*\*\*  
## hr20 158.1195 7.0488 22.432 < 2e-16 \*\*\*  
## hr21 107.9022 6.9453 15.536 < 2e-16 \*\*\*  
## hr22 72.0674 6.9890 10.312 < 2e-16 \*\*\*  
## hr23 31.3404 7.0004 4.477 7.64e-06 \*\*\*  
## holidayHoliday -25.5839 6.3712 -4.016 5.97e-05 \*\*\*  
## weekdaySunday -12.8572 3.7603 -3.419 0.00063 \*\*\*  
## weekdayMonday -8.6638 3.8974 -2.223 0.02623 \*   
## weekdayTuesday -6.7687 3.8295 -1.768 0.07716 .   
## weekdayWednesday -3.6852 3.8010 -0.970 0.33231   
## weekdayThursday -3.1739 3.8047 -0.834 0.40418   
## weekdayFriday 0.5683 3.7761 0.151 0.88036   
## temp 293.4586 12.1953 24.063 < 2e-16 \*\*\*  
## weathersitMisty -19.7902 2.3715 -8.345 < 2e-16 \*\*\*  
## weathersitLightPrecip -92.1438 3.8276 -24.073 < 2e-16 \*\*\*  
## weathersitHeavyPrecip -78.2430 64.7522 -1.208 0.22694   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 111.8 on 12114 degrees of freedom  
## Multiple R-squared: 0.6224, Adjusted R-squared: 0.6209   
## F-statistic: 416.1 on 48 and 12114 DF, p-value: < 2.2e-16

# Adjusted R-squared is 0.6255 which indicates reasonable fit.

###### Task 4.

lm\_fit %>% predict(train) %>% bind\_cols(train) %>% metrics(truth = count, estimate = .pred)

## # A tibble: 3 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 rmse standard 112.   
## 2 rsq standard 0.622  
## 3 mae standard 80.9

ggplot(predict\_train, aes(x= .estimate )) + geom\_histogram() + theme\_bw()

## Error in ggplot(predict\_train, aes(x = .estimate)): object 'predict\_train' not found

###### Task 5.

# Determine the R-squared value on the “test”

lm\_fit %>% predict(test) %>% bind\_cols(test) %>% metrics(truth = count, estimate = .pred)

## # A tibble: 3 x 3  
## .metric .estimator .estimate  
## <chr> <chr> <dbl>  
## 1 rmse standard 110.   
## 2 rsq standard 0.627  
## 3 mae standard 80.1

rsq standard 0.616

# The R-squared of the “test” is slightly lower than the “train”, 0.616 versus 0.6255. This is an indication of good model.