MultLinear Regression Assignment

Natasha Minott

1/28/2021

## Module 2 - Assignment 2

## Minott, Natasha

## Multiple Linear Regression Assignment

## Libraries

library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.0 ──

## ✓ ggplot2 3.3.2 ✓ purrr 0.3.4  
## ✓ tibble 3.0.4 ✓ dplyr 1.0.2  
## ✓ tidyr 1.1.2 ✓ stringr 1.4.0  
## ✓ readr 1.4.0 ✓ forcats 0.5.0

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

library(tidymodels)

## ── Attaching packages ────────────────────────────────────── tidymodels 0.1.2 ──

## ✓ broom 0.7.2 ✓ recipes 0.1.15  
## ✓ dials 0.0.9 ✓ rsample 0.0.8   
## ✓ infer 0.5.4 ✓ tune 0.1.2   
## ✓ modeldata 0.1.0 ✓ workflows 0.2.1   
## ✓ parsnip 0.1.5 ✓ yardstick 0.0.7

## ── 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()

library(glmnet)

## Loading required package: Matrix

##   
## Attaching package: 'Matrix'

## The following objects are masked from 'package:tidyr':  
##   
## expand, pack, unpack

## Loaded glmnet 4.1

library(GGally)

## Registered S3 method overwritten by 'GGally':  
## method from   
## +.gg ggplot2

library(ggcorrplot)  
library(MASS)

##   
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':  
##   
## select

library(car)

## Loading required package: carData

##   
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':  
##   
## recode

## The following object is masked from 'package:purrr':  
##   
## some

library(lubridate)

##   
## Attaching package: 'lubridate'

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

library(lmtest)

## Loading required package: zoo

##   
## Attaching package: 'zoo'

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

## Task 1

bike = bike\_cleaned <- read\_csv("~/Desktop/502 Predictive Analytics/bike\_cleaned.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()  
## )

bike = bike %>% mutate(dteday = mdy(dteday)) #Convert “dteday” from a character variable to a date variable.   
str(bike)

## tibble [17,379 × 16] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ instant : num [1:17379] 1 2 3 4 5 6 7 8 9 10 ...  
## $ dteday : Date[1:17379], format: "2011-01-01" "2011-01-01" ...  
## $ season : chr [1:17379] "Winter" "Winter" "Winter" "Winter" ...  
## $ mnth : chr [1:17379] "Jan" "Jan" "Jan" "Jan" ...  
## $ hr : num [1:17379] 0 1 2 3 4 5 6 7 8 9 ...  
## $ holiday : chr [1:17379] "NotHoliday" "NotHoliday" "NotHoliday" "NotHoliday" ...  
## $ weekday : chr [1:17379] "Saturday" "Saturday" "Saturday" "Saturday" ...  
## $ workingday: chr [1:17379] "NotWorkingDay" "NotWorkingDay" "NotWorkingDay" "NotWorkingDay" ...  
## $ weathersit: chr [1:17379] "NoPrecip" "NoPrecip" "NoPrecip" "NoPrecip" ...  
## $ temp : num [1:17379] 0.24 0.22 0.22 0.24 0.24 0.24 0.22 0.2 0.24 0.32 ...  
## $ atemp : num [1:17379] 0.288 0.273 0.273 0.288 0.288 ...  
## $ hum : num [1:17379] 0.81 0.8 0.8 0.75 0.75 0.75 0.8 0.86 0.75 0.76 ...  
## $ windspeed : num [1:17379] 0 0 0 0 0 0.0896 0 0 0 0 ...  
## $ casual : num [1:17379] 3 8 5 3 0 0 2 1 1 8 ...  
## $ registered: num [1:17379] 13 32 27 10 1 1 0 2 7 6 ...  
## $ count : num [1:17379] 16 40 32 13 1 1 2 3 8 14 ...  
## - attr(\*, "spec")=  
## .. 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()  
## .. )

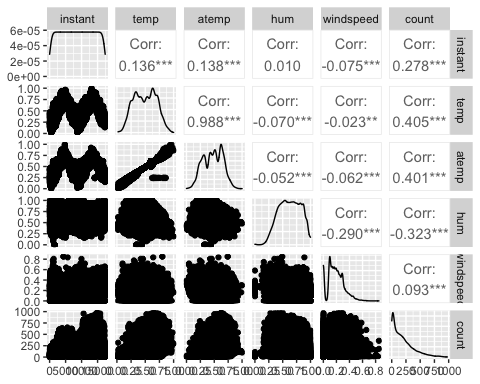
bike = bike %>% mutate\_if(is.character,as\_factor) # to change all characters to factors  
bike = bike %>% mutate(hr = as\_factor(hr)) # to change hr to factor because it is a categorical variable, not necessarily a number that is meant to be calculated   
str(bike)

## tibble [17,379 × 16] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ instant : num [1:17379] 1 2 3 4 5 6 7 8 9 10 ...  
## $ dteday : Date[1:17379], 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 2 3 4 5 6 7 8 9 10 ...  
## $ 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 1 2 1 1 1 1 ...  
## $ temp : num [1:17379] 0.24 0.22 0.22 0.24 0.24 0.24 0.22 0.2 0.24 0.32 ...  
## $ atemp : num [1:17379] 0.288 0.273 0.273 0.288 0.288 ...  
## $ hum : num [1:17379] 0.81 0.8 0.8 0.75 0.75 0.75 0.8 0.86 0.75 0.76 ...  
## $ windspeed : num [1:17379] 0 0 0 0 0 0.0896 0 0 0 0 ...  
## $ casual : num [1:17379] 3 8 5 3 0 0 2 1 1 8 ...  
## $ registered: num [1:17379] 13 32 27 10 1 1 0 2 7 6 ...  
## $ count : num [1:17379] 16 40 32 13 1 1 2 3 8 14 ...  
## - attr(\*, "spec")=  
## .. 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()  
## .. )

**We change hr to factor because it is a categorical variable, not numeric.**

## Task 2

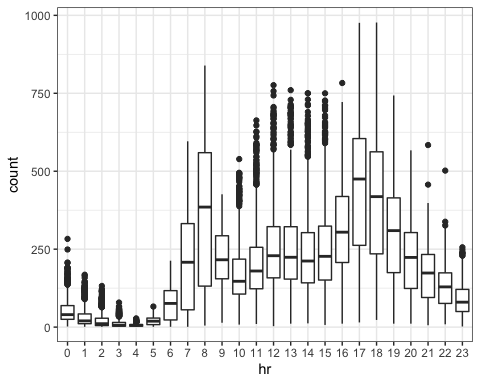
ggpairs(bike, columns = c(1,10,11,12,13,16))#using correlation to see relationships.



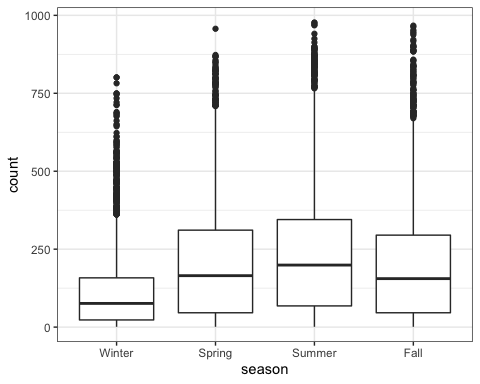
**Temperature seems to be the best correlated with count.**

## Task 3

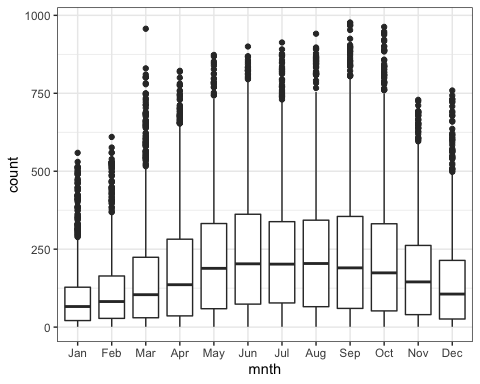
ggplot(bike,aes(x=hr,y=count)) + geom\_boxplot() + theme\_bw()



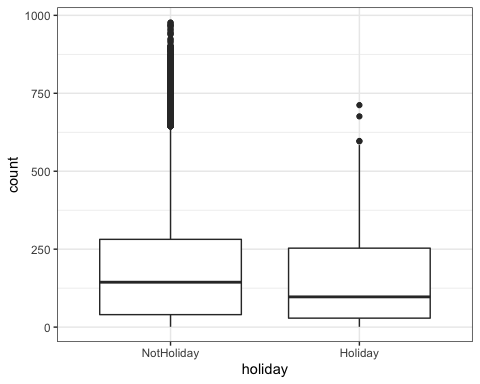
ggplot(bike,aes(x=season,y=count)) + geom\_boxplot() + theme\_bw()



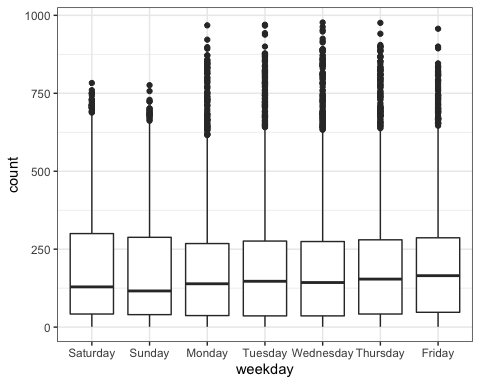
ggplot(bike,aes(x=mnth,y=count)) + geom\_boxplot() + theme\_bw()



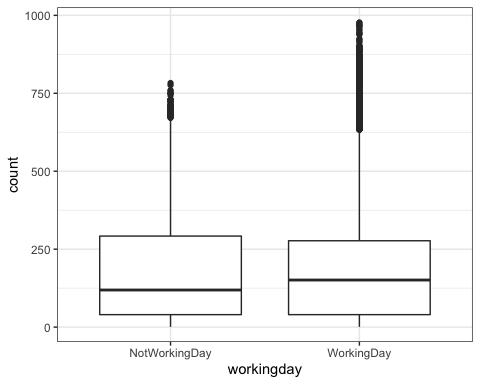
ggplot(bike,aes(x=holiday,y=count)) + geom\_boxplot() + theme\_bw()



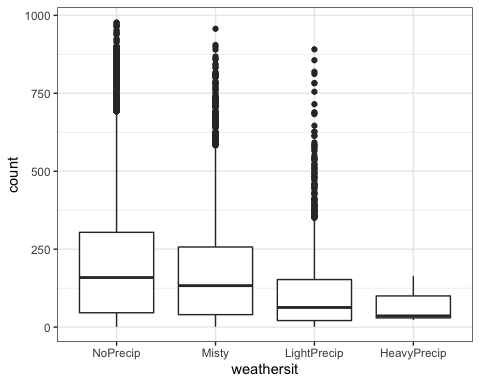
ggplot(bike,aes(x=weekday,y=count)) + geom\_boxplot() + theme\_bw()



ggplot(bike,aes(x=workingday,y=count)) + geom\_boxplot() +theme\_bw()



ggplot(bike,aes(x=weathersit,y=count)) + geom\_boxplot() + theme\_bw()

 **Month,season, weathersit, and working day clearly affect count. From looking at our quantitative variables, there is a strong correlation between temp and count. The month season and weathersit are variables play of eachother since they are all related to weather.It is important to note there there are many outliers for season For the variable working day, people may use a bike to commute to work.**

## Task 4

bike\_count\_recipe = recipe(count ~ hr, bike) # model using hr gives us Adjusted Rsquare value of 0.5015  
  
  
lm\_model =   
 linear\_reg()%>%  
 set\_engine("lm")  
   
 lm\_wflow=  
 workflow()%>%  
 add\_model(lm\_model)%>%  
 add\_recipe(bike\_count\_recipe)  
   
 lm\_fit = fit(lm\_wflow, bike)

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

##   
## Call:  
## stats::lm(formula = ..y ~ ., data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -446.45 -60.99 -6.01 50.10 551.49   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 53.898 4.756 11.332 < 2e-16 \*\*\*  
## hr1 -20.522 6.731 -3.049 0.002300 \*\*   
## hr2 -31.028 6.752 -4.595 4.35e-06 \*\*\*  
## hr3 -42.171 6.796 -6.205 5.58e-10 \*\*\*  
## hr4 -47.545 6.796 -6.996 2.73e-12 \*\*\*  
## hr5 -34.008 6.747 -5.040 4.70e-07 \*\*\*  
## hr6 22.146 6.729 3.291 0.000999 \*\*\*  
## hr7 158.167 6.724 23.523 < 2e-16 \*\*\*  
## hr8 305.113 6.724 45.377 < 2e-16 \*\*\*  
## hr9 165.411 6.724 24.600 < 2e-16 \*\*\*  
## hr10 119.770 6.724 17.812 < 2e-16 \*\*\*  
## hr11 154.245 6.724 22.939 < 2e-16 \*\*\*  
## hr12 199.418 6.722 29.668 < 2e-16 \*\*\*  
## hr13 199.763 6.719 29.729 < 2e-16 \*\*\*  
## hr14 187.051 6.719 27.838 < 2e-16 \*\*\*  
## hr15 197.335 6.719 29.368 < 2e-16 \*\*\*  
## hr16 258.085 6.717 38.422 < 2e-16 \*\*\*  
## hr17 407.554 6.717 60.674 < 2e-16 \*\*\*  
## hr18 371.613 6.722 55.286 < 2e-16 \*\*\*  
## hr19 257.625 6.722 38.327 < 2e-16 \*\*\*  
## hr20 172.132 6.722 25.608 < 2e-16 \*\*\*  
## hr21 118.416 6.722 17.617 < 2e-16 \*\*\*  
## hr22 77.437 6.722 11.520 < 2e-16 \*\*\*  
## hr23 33.933 6.722 5.048 4.50e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 128.2 on 17355 degrees of freedom  
## Multiple R-squared: 0.5015, Adjusted R-squared: 0.5008   
## F-statistic: 759.1 on 23 and 17355 DF, p-value: < 2.2e-16

**For this task I used Hours to predict count and ended up with an R-squared value of 0.5015 which indicates that this is a decent model. When I plugged month, season, or workingday into the model, r squared values that were less than 0.5 were being displayed.**

## Task 5

bike2 = bike %>% dplyr::select("season", "mnth", "hr", "holiday", "weekday", "workingday", "weathersit", "temp", "atemp","hum","windspeed","count") #selecting appropriate values.  
summary(bike2)

## season mnth hr holiday   
## Winter:4242 May :1488 16 : 730 NotHoliday:16879   
## Spring:4409 Jul :1488 17 : 730 Holiday : 500   
## Summer:4496 Dec :1483 13 : 729   
## Fall :4232 Aug :1475 14 : 729   
## Mar :1473 15 : 729   
## Oct :1451 12 : 728   
## (Other):8521 (Other):13004   
## weekday workingday weathersit temp   
## Saturday :2512 NotWorkingDay: 5514 NoPrecip :11413 Min. :0.020   
## Sunday :2502 WorkingDay :11865 Misty : 4544 1st Qu.:0.340   
## Monday :2479 LightPrecip: 1419 Median :0.500   
## Tuesday :2453 HeavyPrecip: 3 Mean :0.497   
## Wednesday:2475 3rd Qu.:0.660   
## Thursday :2471 Max. :1.000   
## Friday :2487   
## atemp hum windspeed count   
## Min. :0.0000 Min. :0.0000 Min. :0.0000 Min. : 1.0   
## 1st Qu.:0.3333 1st Qu.:0.4800 1st Qu.:0.1045 1st Qu.: 40.0   
## Median :0.4848 Median :0.6300 Median :0.1940 Median :142.0   
## Mean :0.4758 Mean :0.6272 Mean :0.1901 Mean :189.5   
## 3rd Qu.:0.6212 3rd Qu.:0.7800 3rd Qu.:0.2537 3rd Qu.:281.0   
## Max. :1.0000 Max. :1.0000 Max. :0.8507 Max. :977.0   
##

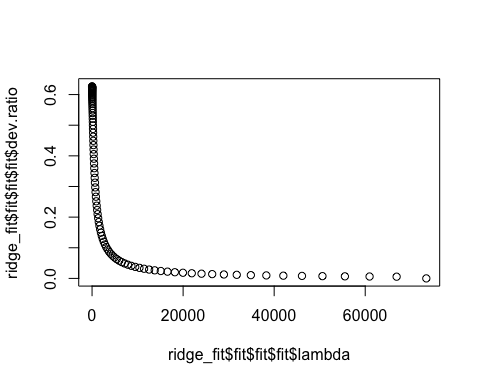
glimpse(bike2)

## Rows: 17,379  
## Columns: 12  
## $ season <fct> Winter, Winter, Winter, Winter, Winter, Winter, Winter, Wi…  
## $ mnth <fct> Jan, Jan, Jan, Jan, Jan, Jan, Jan, Jan, Jan, Jan, Jan, Jan…  
## $ hr <fct> 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, …  
## $ holiday <fct> NotHoliday, NotHoliday, NotHoliday, NotHoliday, NotHoliday…  
## $ weekday <fct> Saturday, Saturday, Saturday, Saturday, Saturday, Saturday…  
## $ workingday <fct> NotWorkingDay, NotWorkingDay, NotWorkingDay, NotWorkingDay…  
## $ weathersit <fct> NoPrecip, NoPrecip, NoPrecip, NoPrecip, NoPrecip, Misty, N…  
## $ temp <dbl> 0.24, 0.22, 0.22, 0.24, 0.24, 0.24, 0.22, 0.20, 0.24, 0.32…  
## $ atemp <dbl> 0.2879, 0.2727, 0.2727, 0.2879, 0.2879, 0.2576, 0.2727, 0.…  
## $ hum <dbl> 0.81, 0.80, 0.80, 0.75, 0.75, 0.75, 0.80, 0.86, 0.75, 0.76…  
## $ windspeed <dbl> 0.0000, 0.0000, 0.0000, 0.0000, 0.0000, 0.0896, 0.0000, 0.…  
## $ count <dbl> 16, 40, 32, 13, 1, 1, 2, 3, 8, 14, 36, 56, 84, 94, 106, 11…

bike2\_recipe= recipe(count ~ season +mnth + hr + holiday + weekday + workingday + weathersit + temp + atemp + hum + windspeed, bike2) %>%  
 step\_other(hr, season, mnth,holiday,weekday, threshold = 0.01)%>%  
 step\_dummy(all\_nominal())%>%  
 step\_center(all\_predictors())%>%  
 step\_scale(all\_predictors())  
  
ridge\_model=  
 linear\_reg(mixture=0)%>%  
 set\_engine("glmnet")  
  
ridge\_wflow =   
 workflow()%>%  
 add\_model(ridge\_model)%>%  
 add\_recipe(bike2\_recipe)  
  
ridge\_fit = fit(ridge\_wflow, bike2)  
  
ridge\_fit

## ══ Workflow [trained] ══════════════════════════════════════════════════════════  
## Preprocessor: Recipe  
## Model: linear\_reg()  
##   
## ── Preprocessor ────────────────────────────────────────────────────────────────  
## 4 Recipe Steps  
##   
## ● step\_other()  
## ● step\_dummy()  
## ● step\_center()  
## ● step\_scale()  
##   
## ── Model ───────────────────────────────────────────────────────────────────────  
##   
## Call: glmnet::glmnet(x = maybe\_matrix(x), y = y, family = "gaussian", alpha = ~0)   
##   
## Df %Dev Lambda  
## 1 52 0.00 73420  
## 2 52 0.56 66900  
## 3 52 0.61 60950  
## 4 52 0.67 55540  
## 5 52 0.74 50600  
## 6 52 0.81 46110  
## 7 52 0.89 42010  
## 8 52 0.97 38280  
## 9 52 1.07 34880  
## 10 52 1.17 31780  
## 11 52 1.28 28960  
## 12 52 1.40 26390  
## 13 52 1.54 24040  
## 14 52 1.68 21910  
## 15 52 1.84 19960  
## 16 52 2.01 18190  
## 17 52 2.20 16570  
## 18 52 2.41 15100  
## 19 52 2.64 13760  
## 20 52 2.88 12540  
## 21 52 3.15 11420  
## 22 52 3.44 10410  
## 23 52 3.75 9482  
## 24 52 4.10 8640  
## 25 52 4.47 7872  
## 26 52 4.87 7173  
## 27 52 5.31 6536  
## 28 52 5.78 5955  
## 29 52 6.29 5426  
## 30 52 6.83 4944  
## 31 52 7.42 4505  
## 32 52 8.06 4105  
## 33 52 8.73 3740  
## 34 52 9.46 3408  
## 35 52 10.24 3105  
## 36 52 11.07 2829  
## 37 52 11.95 2578  
## 38 52 12.88 2349  
## 39 52 13.88 2140  
## 40 52 14.92 1950  
## 41 52 16.02 1777  
## 42 52 17.18 1619  
## 43 52 18.39 1475  
## 44 52 19.65 1344  
## 45 52 20.96 1225  
## 46 52 22.32 1116  
##   
## ...  
## and 54 more lines.

plot(ridge\_fit$fit$fit$fit$lambda,ridge\_fit$fit$fit$fit$dev.ratio)



ridge\_fit%>%  
 pull\_workflow\_fit()%>%  
 pluck("fit")

##   
## Call: glmnet::glmnet(x = maybe\_matrix(x), y = y, family = "gaussian", alpha = ~0)   
##   
## Df %Dev Lambda  
## 1 52 0.00 73420  
## 2 52 0.56 66900  
## 3 52 0.61 60950  
## 4 52 0.67 55540  
## 5 52 0.74 50600  
## 6 52 0.81 46110  
## 7 52 0.89 42010  
## 8 52 0.97 38280  
## 9 52 1.07 34880  
## 10 52 1.17 31780  
## 11 52 1.28 28960  
## 12 52 1.40 26390  
## 13 52 1.54 24040  
## 14 52 1.68 21910  
## 15 52 1.84 19960  
## 16 52 2.01 18190  
## 17 52 2.20 16570  
## 18 52 2.41 15100  
## 19 52 2.64 13760  
## 20 52 2.88 12540  
## 21 52 3.15 11420  
## 22 52 3.44 10410  
## 23 52 3.75 9482  
## 24 52 4.10 8640  
## 25 52 4.47 7872  
## 26 52 4.87 7173  
## 27 52 5.31 6536  
## 28 52 5.78 5955  
## 29 52 6.29 5426  
## 30 52 6.83 4944  
## 31 52 7.42 4505  
## 32 52 8.06 4105  
## 33 52 8.73 3740  
## 34 52 9.46 3408  
## 35 52 10.24 3105  
## 36 52 11.07 2829  
## 37 52 11.95 2578  
## 38 52 12.88 2349  
## 39 52 13.88 2140  
## 40 52 14.92 1950  
## 41 52 16.02 1777  
## 42 52 17.18 1619  
## 43 52 18.39 1475  
## 44 52 19.65 1344  
## 45 52 20.96 1225  
## 46 52 22.32 1116  
## 47 52 23.73 1017  
## 48 52 25.17 926  
## 49 52 26.65 844  
## 50 52 28.16 769  
## 51 52 29.70 701  
## 52 52 31.25 639  
## 53 52 32.82 582  
## 54 52 34.39 530  
## 55 52 35.96 483  
## 56 52 37.51 440  
## 57 52 39.06 401  
## 58 52 40.57 365  
## 59 52 42.06 333  
## 60 52 43.50 303  
## 61 52 44.90 276  
## 62 52 46.25 252  
## 63 52 47.55 230  
## 64 52 48.78 209  
## 65 52 49.95 190  
## 66 52 51.06 174  
## 67 52 52.10 158  
## 68 52 53.07 144  
## 69 52 53.97 131  
## 70 52 54.80 120  
## 71 52 55.57 109  
## 72 52 56.28 99  
## 73 52 56.92 91  
## 74 52 57.50 82  
## 75 52 58.03 75  
## 76 52 58.51 68  
## 77 52 58.94 62  
## 78 52 59.33 57  
## 79 52 59.68 52  
## 80 52 60.00 47  
## 81 52 60.28 43  
## 82 52 60.53 39  
## 83 52 60.76 36  
## 84 52 60.96 33  
## 85 52 61.15 30  
## 86 52 61.31 27  
## 87 52 61.47 25  
## 88 52 61.61 22  
## 89 52 61.73 20  
## 90 52 61.85 19  
## 91 52 61.96 17  
## 92 52 62.06 15  
## 93 52 62.16 14  
## 94 52 62.24 13  
## 95 52 62.33 12  
## 96 52 62.41 11  
## 97 52 62.48 10  
## 98 52 62.54 9  
## 99 52 62.61 8  
## 100 52 62.67 7

ridge\_fit%>%  
 pull\_workflow\_fit()%>%  
 pluck("fit") %>%  
 coef(s=13) #this is the coefficient for my selected lambda value

## 53 x 1 sparse Matrix of class "dgCMatrix"  
## 1  
## (Intercept) 189.4630876  
## temp 28.4366729  
## atemp 26.6537839  
## hum -24.4557593  
## windspeed -3.2276179  
## season\_Spring 10.0892335  
## season\_Summer 3.5388436  
## season\_Fall 19.4971495  
## mnth\_Feb -1.1205294  
## mnth\_Mar 1.6701576  
## mnth\_Apr 0.7531169  
## mnth\_May 3.9377605  
## mnth\_Jun -1.2860739  
## mnth\_Jul -5.6065875  
## mnth\_Aug 1.1769839  
## mnth\_Sep 9.1250536  
## mnth\_Oct 5.8072456  
## mnth\_Nov 0.5414665  
## mnth\_Dec 0.3570480  
## hr\_X1 -17.2361977  
## hr\_X2 -18.5649284  
## hr\_X3 -20.1161969  
## hr\_X4 -20.4280957  
## hr\_X5 -17.6649021  
## hr\_X6 -6.8380221  
## hr\_X7 18.2005321  
## hr\_X8 44.1092996  
## hr\_X9 15.8610480  
## hr\_X10 5.0301781  
## hr\_X11 9.2069250  
## hr\_X12 16.0731648  
## hr\_X13 14.8202446  
## hr\_X14 11.6307273  
## hr\_X15 13.3428717  
## hr\_X16 25.0999101  
## hr\_X17 54.2019963  
## hr\_X18 48.4809099  
## hr\_X19 28.6256311  
## hr\_X20 14.1045184  
## hr\_X21 5.2283682  
## hr\_X22 -1.4023523  
## hr\_X23 -8.4521997  
## holiday\_Holiday -4.3729416  
## weekday\_Sunday -4.3754849  
## weekday\_Monday -1.5823982  
## weekday\_Tuesday -1.0944100  
## weekday\_Wednesday -0.1016868  
## weekday\_Thursday -0.5675788  
## weekday\_Friday 0.9182103  
## workingday\_WorkingDay -0.2130578  
## weathersit\_Misty -1.6741479  
## weathersit\_LightPrecip -14.3539746  
## weathersit\_HeavyPrecip -0.2935185

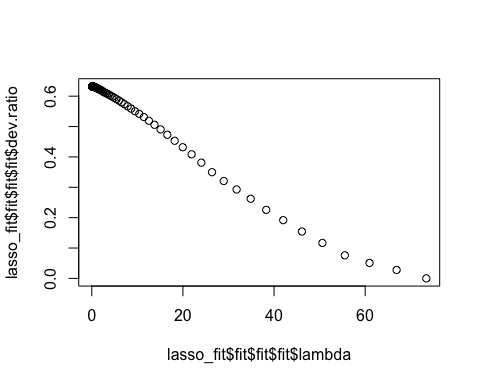
**The ridge method got us a rsquared value of .6224 which is higher than my previous models where I only used one variable to predict count.**

## Task 6

bike2\_recipe= recipe(count ~ season +mnth + hr + holiday + weekday + workingday + weathersit + temp + atemp + hum + windspeed, bike2) %>%  
 step\_other(hr, season, mnth,holiday,weekday, threshold = 0.01)%>%  
 step\_dummy(all\_nominal())%>%  
 step\_center(all\_predictors())%>%  
 step\_scale(all\_predictors())  
  
lasso\_model=  
 linear\_reg(mixture=1)%>%  
 set\_engine("glmnet")  
  
lasso\_wflow =   
 workflow()%>%  
 add\_model(lasso\_model)%>%  
 add\_recipe(bike2\_recipe)  
  
lasso\_fit = fit(lasso\_wflow, bike2)  
  
lasso\_fit %>%  
 pull\_workflow\_fit()%>%  
 pluck("fit")

##   
## Call: glmnet::glmnet(x = maybe\_matrix(x), y = y, family = "gaussian", alpha = ~1)   
##   
## Df %Dev Lambda  
## 1 0 0.00 73.420  
## 2 1 2.78 66.900  
## 3 1 5.09 60.950  
## 4 3 7.60 55.540  
## 5 3 11.69 50.600  
## 6 4 15.44 46.110  
## 7 4 19.18 42.010  
## 8 6 22.56 38.280  
## 9 6 26.23 34.880  
## 10 6 29.28 31.780  
## 11 8 32.06 28.960  
## 12 11 34.97 26.390  
## 13 12 38.11 24.040  
## 14 12 40.86 21.910  
## 15 13 43.19 19.960  
## 16 14 45.32 18.190  
## 17 15 47.30 16.570  
## 18 15 49.05 15.100  
## 19 16 50.59 13.760  
## 20 17 51.90 12.540  
## 21 18 53.13 11.420  
## 22 18 54.16 10.410  
## 23 19 55.02 9.482  
## 24 22 55.90 8.640  
## 25 23 56.68 7.872  
## 26 25 57.37 7.173  
## 27 26 58.00 6.536  
## 28 27 58.56 5.955  
## 29 27 59.04 5.426  
## 30 30 59.47 4.944  
## 31 31 59.86 4.505  
## 32 32 60.19 4.105  
## 33 32 60.51 3.740  
## 34 33 60.79 3.408  
## 35 33 61.02 3.105  
## 36 33 61.20 2.829  
## 37 34 61.37 2.578  
## 38 37 61.65 2.349  
## 39 37 61.86 2.140  
## 40 37 62.03 1.950  
## 41 38 62.16 1.777  
## 42 38 62.27 1.619  
## 43 38 62.37 1.475  
## 44 41 62.46 1.344  
## 45 41 62.58 1.225  
## 46 42 62.67 1.116  
## 47 42 62.76 1.017  
## 48 41 62.81 0.926  
## 49 42 62.86 0.844  
## 50 43 62.90 0.769  
## 51 43 62.94 0.701  
## 52 44 62.97 0.639  
## 53 43 63.01 0.582  
## 54 44 63.03 0.530  
## 55 44 63.05 0.483  
## 56 43 63.07 0.440  
## 57 44 63.09 0.401  
## 58 46 63.11 0.365  
## 59 47 63.13 0.333  
## 60 48 63.14 0.303  
## 61 48 63.15 0.276  
## 62 48 63.16 0.252  
## 63 48 63.17 0.230  
## 64 48 63.18 0.209  
## 65 48 63.19 0.190  
## 66 48 63.19 0.174  
## 67 49 63.20 0.158  
## 68 49 63.20 0.144  
## 69 49 63.20 0.131  
## 70 51 63.21 0.120  
## 71 51 63.21 0.109  
## 72 51 63.21 0.099  
## 73 51 63.21 0.091  
## 74 51 63.22 0.082  
## 75 51 63.22 0.075  
## 76 51 63.22 0.068  
## 77 51 63.22 0.062

plot(lasso\_fit$fit$fit$fit$lambda,lasso\_fit$fit$fit$fit$dev.ratio)



lasso\_fit%>%  
 pull\_workflow\_fit()%>%  
 pluck("fit") %>%  
 coef(s= 0.099) #this is the coefficient for my selected lambda value

## 53 x 1 sparse Matrix of class "dgCMatrix"  
## 1  
## (Intercept) 189.46308763  
## temp 36.06274758  
## atemp 18.70995826  
## hum -22.38747924  
## windspeed -5.06259899  
## season\_Spring 14.35923553  
## season\_Summer 8.50999962  
## season\_Fall 25.72565709  
## mnth\_Feb 0.04739374  
## mnth\_Mar 2.14575435  
## mnth\_Apr -0.22978117  
## mnth\_May 2.47554784  
## mnth\_Jun -2.87617713  
## mnth\_Jul -7.89854414  
## mnth\_Aug -0.88481075  
## mnth\_Sep 7.17110327  
## mnth\_Oct 3.17503422  
## mnth\_Nov -2.16819456  
## mnth\_Dec -1.08923230  
## hr\_X1 -4.74367545  
## hr\_X2 -6.28352732  
## hr\_X3 -8.15080249  
## hr\_X4 -8.48710688  
## hr\_X5 -5.37445742  
## hr\_X6 6.10817662  
## hr\_X7 32.96365315  
## hr\_X8 60.74438657  
## hr\_X9 30.68626876  
## hr\_X10 19.22161540  
## hr\_X11 23.78975488  
## hr\_X12 31.25416249  
## hr\_X13 29.98969087  
## hr\_X14 26.64074419  
## hr\_X15 28.48331602  
## hr\_X16 41.03764952  
## hr\_X17 72.09662498  
## hr\_X18 65.89148378  
## hr\_X19 44.54369262  
## hr\_X20 28.93486291  
## hr\_X21 19.33535358  
## hr\_X22 12.17666238  
## hr\_X23 4.59336457  
## holiday\_Holiday -4.63070131  
## weekday\_Sunday -4.84534308  
## weekday\_Monday -1.42426998  
## weekday\_Tuesday -0.95459732  
## weekday\_Wednesday .   
## weekday\_Thursday -0.34986593  
## weekday\_Friday 1.03749031  
## workingday\_WorkingDay -0.75801419  
## weathersit\_Misty -2.54315297  
## weathersit\_LightPrecip -16.31480945  
## weathersit\_HeavyPrecip -0.31616793

**With the lasso method, Wednesday fell off the model and we now have an rsquared value of 0.6321 which is higher than my ridge model that gave me an rsquared value of 0.6224**