Lab\_Three\_Salameh

Sief Salameh

5/8/2023

# Part 3 - Data Analysis

setwd("~/Downloads/Machine\_Learning/Lab\_Three\_Salameh")  
  
library(tidyverse)

## Warning: package 'ggplot2' was built under R version 4.1.2

## Warning: package 'tibble' was built under R version 4.1.2

## Warning: package 'tidyr' was built under R version 4.1.2

## Warning: package 'readr' was built under R version 4.1.2

## Warning: package 'purrr' was built under R version 4.1.2

## Warning: package 'dplyr' was built under R version 4.1.2

## Warning: package 'stringr' was built under R version 4.1.2

library(gbm)

## Warning: package 'gbm' was built under R version 4.1.2

library(ISLR)  
library(tree)

## Warning: package 'tree' was built under R version 4.1.2

library(glmnet)

## Warning: package 'glmnet' was built under R version 4.1.2

library(leaps)  
  
Covid\_df <- read.csv("CovidData.csv")

# Question 1:

The “VariableDescription.xlsx” spreadsheet contains a list of variables that we’ll use for our analyses. Note that this is not a full list of all the variables in the dataset, although it’s close (we ignoring a few perfectly co-linear predictors). Filter the full set of variables in the dataset down to the Opportunity Insights and PM COVID variables listed in the spreadsheet along with ‘county’, ‘state’ and ‘deathspc’.

Covid\_df <- Covid\_df[, c(  
 "state", "deathspc", "intersects\_msa", "cur\_smoke\_q1", "cur\_smoke\_q2", "cur\_smoke\_q3", "cur\_smoke\_q4", "bmi\_obese\_q1", "bmi\_obese\_q2", "bmi\_obese\_q3", "bmi\_obese\_q4", "exercise\_any\_q1",  
 "exercise\_any\_q2", "exercise\_any\_q3", "exercise\_any\_q4", "brfss\_mia", "puninsured2010", "reimb\_penroll\_adj10", "mort\_30day\_hosp\_z", "adjmortmeas\_amiall30day", "adjmortmeas\_chfall30day", "med\_prev\_qual\_z", "primcarevis\_10", "diab\_hemotest\_10", "diab\_eyeexam\_10", "diab\_lipids\_10", "mammogram\_10", "cs00\_seg\_inc", "cs00\_seg\_inc\_pov25", "cs00\_seg\_inc\_aff75", "cs\_race\_theil\_2000", "gini99", "poor\_share", "inc\_share\_1perc",  
 "frac\_middleclass", "scap\_ski90pcm", "rel\_tot", "cs\_frac\_black", "cs\_frac\_hisp", "unemp\_rate", "cs\_labforce", "cs\_elf\_ind\_man", "cs\_born\_foreign", "mig\_inflow", "mig\_outflow", "pop\_density", "frac\_traveltime\_lt15", "hhinc00", "median\_house\_value", "ccd\_exp\_tot", "score\_r", "cs\_fam\_wkidsinglemom", "subcty\_exp\_pc", "taxrate", "tax\_st\_diff\_top20", "pm25", "pm25\_mia",  
 "summer\_tmmx", "summer\_rmax", "winter\_tmmx", "winter\_rmax", "bmcruderate"  
)]

# Question 2:

Compute descriptive (summary) statistics for the subset of Opportunity Insights and PM COVID variables you filtered in previous question.

summary(Covid\_df)

## state deathspc intersects\_msa cur\_smoke\_q1   
## Length:3107 Min. : 0.000 Min. :0.0000 Min. :0.0000   
## Class :character 1st Qu.: 0.000 1st Qu.:0.0000 1st Qu.:0.0000   
## Mode :character Median : 3.802 Median :1.0000 Median :0.2500   
## Mean : 23.790 Mean :0.5967 Mean :0.2127   
## 3rd Qu.: 21.462 3rd Qu.:1.0000 3rd Qu.:0.3109   
## Max. :2279.611 Max. :1.0000 Max. :1.0000   
##   
## cur\_smoke\_q2 cur\_smoke\_q3 cur\_smoke\_q4 bmi\_obese\_q1   
## Min. :0.0000 Min. :0.0000 Min. :0.00000 Min. :0.00000   
## 1st Qu.:0.0000 1st Qu.:0.0000 1st Qu.:0.00000 1st Qu.:0.08013   
## Median :0.1987 Median :0.1429 Median :0.09653 Median :0.27208   
## Mean :0.1710 Mean :0.1345 Mean :0.09832 Mean :0.23917   
## 3rd Qu.:0.2500 3rd Qu.:0.2000 3rd Qu.:0.14872 3rd Qu.:0.33553   
## Max. :1.0000 Max. :1.0000 Max. :1.00000 Max. :1.00000   
##   
## bmi\_obese\_q2 bmi\_obese\_q3 bmi\_obese\_q4 exercise\_any\_q1   
## Min. :0.0000 Min. :0.0000 Min. :0.0000 Min. :0.0000   
## 1st Qu.:0.0000 1st Qu.:0.0000 1st Qu.:0.0000 1st Qu.:0.3125   
## Median :0.2416 Median :0.2231 Median :0.1941 Median :0.5666   
## Mean :0.2146 Mean :0.2096 Mean :0.1867 Mean :0.4560   
## 3rd Qu.:0.3043 3rd Qu.:0.2972 3rd Qu.:0.2667 3rd Qu.:0.6415   
## Max. :1.0000 Max. :1.0000 Max. :1.0000 Max. :1.0000   
##   
## exercise\_any\_q2 exercise\_any\_q3 exercise\_any\_q4 brfss\_mia   
## Min. :0.0000 Min. :0.0000 Min. :0.0000 Min. :0.0000   
## 1st Qu.:0.4444 1st Qu.:0.3542 1st Qu.:0.4000 1st Qu.:0.0000   
## Median :0.7071 Median :0.7784 Median :0.8333 Median :0.0000   
## Mean :0.5557 Mean :0.6038 Mean :0.6387 Mean :0.2494   
## 3rd Qu.:0.7692 3rd Qu.:0.8418 3rd Qu.:0.8905 3rd Qu.:0.0000   
## Max. :1.0000 Max. :1.0000 Max. :1.0000 Max. :1.0000   
##   
## puninsured2010 reimb\_penroll\_adj10 mort\_30day\_hosp\_z adjmortmeas\_amiall30day  
## Min. : 3.625 Min. : 3664 Min. :-7.7780 Min. :0.0000   
## 1st Qu.:14.410 1st Qu.: 8159 1st Qu.:-0.2559 1st Qu.:0.1453   
## Median :18.147 Median : 9194 Median : 0.4001 Median :0.1627   
## Mean :18.469 Mean : 9303 Mean : 0.4578 Mean :0.1655   
## 3rd Qu.:21.961 3rd Qu.:10285 3rd Qu.: 1.1478 3rd Qu.:0.1834   
## Max. :41.366 Max. :18443 Max. : 8.4727 Max. :0.4447   
## NA's :4 NA's :1 NA's :1   
## adjmortmeas\_chfall30day med\_prev\_qual\_z primcarevis\_10 diab\_hemotest\_10  
## Min. :0.0000 Min. :-4.85385 Min. :18.33 Min. : 16.91   
## 1st Qu.:0.0963 1st Qu.:-0.61559 1st Qu.:78.80 1st Qu.: 81.11   
## Median :0.1072 Median :-0.09023 Median :82.20 Median : 84.78   
## Mean :0.1090 Mean :-0.14855 Mean :80.87 Mean : 83.71   
## 3rd Qu.:0.1202 3rd Qu.: 0.44443 3rd Qu.:84.96 3rd Qu.: 87.68   
## Max. :0.3445 Max. : 3.47852 Max. :95.67 Max. :100.00   
## NA's :95 NA's :9 NA's :38   
## diab\_eyeexam\_10 diab\_lipids\_10 mammogram\_10 cs00\_seg\_inc   
## Min. :31.37 Min. :19.66 Min. :30.00 Min. :-0.013363   
## 1st Qu.:61.26 1st Qu.:75.00 1st Qu.:57.94 1st Qu.: 0.005047   
## Median :65.98 Median :79.76 Median :63.62 Median : 0.013647   
## Mean :66.08 Mean :78.31 Mean :63.11 Mean : 0.025892   
## 3rd Qu.:70.91 3rd Qu.:83.34 3rd Qu.:68.91 3rd Qu.: 0.036453   
## Max. :90.00 Max. :94.48 Max. :95.24 Max. : 0.438241   
## NA's :53 NA's :50 NA's :78   
## cs00\_seg\_inc\_pov25 cs00\_seg\_inc\_aff75 cs\_race\_theil\_2000 gini99   
## Min. :-0.019502 Min. :-0.001993 Min. :0.00000 Min. :0.1610   
## 1st Qu.: 0.004164 1st Qu.: 0.003455 1st Qu.:0.01559 1st Qu.:0.3175   
## Median : 0.013136 Median : 0.012577 Median :0.04719 Median :0.3700   
## Mean : 0.024278 Mean : 0.026463 Mean :0.07540 Mean :0.3790   
## 3rd Qu.: 0.034737 3rd Qu.: 0.037337 3rd Qu.:0.10451 3rd Qu.:0.4295   
## Max. : 0.749106 Max. : 0.196959 Max. :0.71201 Max. :1.0914   
## NA's :99   
## poor\_share inc\_share\_1perc frac\_middleclass scap\_ski90pcm   
## Min. :0.00000 Min. :0.01857 Min. :0.2156 Min. :-4.258739   
## 1st Qu.:0.09538 1st Qu.:0.06258 1st Qu.:0.4919 1st Qu.:-0.964225   
## Median :0.12962 Median :0.08360 Median :0.5598 Median :-0.091105   
## Mean :0.14174 Mean :0.09481 Mean :0.5542 Mean : 0.000182   
## 3rd Qu.:0.17528 3rd Qu.:0.11357 3rd Qu.:0.6228 3rd Qu.: 0.818039   
## Max. :0.56917 Max. :0.73477 Max. :0.8750 Max. : 9.911112   
## NA's :99 NA's :1   
## rel\_tot cs\_frac\_black cs\_frac\_hisp unemp\_rate   
## Min. : 1.816 Min. : 0.0000 Min. : 0.08203 Min. :0.01609   
## 1st Qu.: 39.670 1st Qu.: 0.2645 1st Qu.: 0.91724 1st Qu.:0.03742   
## Median : 51.329 Median : 1.6911 Median : 1.78344 Median :0.04691   
## Mean : 53.225 Mean : 8.7445 Mean : 6.20919 Mean :0.04987   
## 3rd Qu.: 64.787 3rd Qu.:10.0310 3rd Qu.: 5.10768 3rd Qu.:0.05874   
## Max. :164.527 Max. :85.9651 Max. :97.53905 Max. :0.17699   
## NA's :1   
## cs\_labforce cs\_elf\_ind\_man cs\_born\_foreign mig\_inflow   
## Min. :0.3192 Min. :0.00000 Min. : 0.0000 Min. :0.00000   
## 1st Qu.:0.5670 1st Qu.:0.08864 1st Qu.: 0.8985 1st Qu.:0.01650   
## Median :0.6166 Median :0.14939 Median : 1.7273 Median :0.02443   
## Mean :0.6093 Mean :0.15912 Mean : 3.4420 Mean :0.02868   
## 3rd Qu.:0.6580 3rd Qu.:0.21993 3rd Qu.: 3.9221 3rd Qu.:0.03632   
## Max. :0.8609 Max. :0.48554 Max. :50.9357 Max. :0.16867   
## NA's :90   
## mig\_outflow pop\_density frac\_traveltime\_lt15 hhinc00   
## Min. :0.00000 Min. : 0.10 Min. :0.09988 Min. :10512   
## 1st Qu.:0.01877 1st Qu.: 17.48 1st Qu.:0.29993 1st Qu.:28734   
## Median :0.02511 Median : 43.13 Median :0.38582 Median :32235   
## Mean :0.02752 Mean : 244.33 Mean :0.40380 Mean :32854   
## 3rd Qu.:0.03304 3rd Qu.: 104.99 3rd Qu.:0.49909 3rd Qu.:36039   
## Max. :0.15326 Max. :66940.08 Max. :0.81764 Max. :77943   
## NA's :90   
## median\_house\_value ccd\_exp\_tot score\_r cs\_fam\_wkidsinglemom  
## Min. : 0 Min. : 3.032 Min. :-38.68714 Min. :0.02479   
## 1st Qu.: 77047 1st Qu.: 5.027 1st Qu.: -4.96963 1st Qu.:0.15244   
## Median : 100775 Median : 5.785 Median : 0.83494 Median :0.18247   
## Mean : 112180 Mean : 6.093 Mean : 0.07735 Mean :0.19460   
## 3rd Qu.: 128501 3rd Qu.: 6.735 3rd Qu.: 5.99018 3rd Qu.:0.22158   
## Max. :1333001 Max. :53.258 Max. : 32.98522 Max. :0.54388   
## NA's :27 NA's :38   
## subcty\_exp\_pc taxrate tax\_st\_diff\_top20 pm25   
## Min. : 0 Min. :0.00000 Min. :0.0000 Min. : 0.000   
## 1st Qu.: 1510 1st Qu.:0.01499 1st Qu.:0.0000 1st Qu.: 6.310   
## Median : 1936 Median :0.02034 Median :0.0000 Median : 8.785   
## Mean : 2119 Mean :0.02309 Mean :0.7756 Mean : 8.372   
## 3rd Qu.: 2505 3rd Qu.:0.02716 3rd Qu.:1.0000 3rd Qu.:10.484   
## Max. :20542 Max. :0.20991 Max. :7.2200 Max. :15.786   
## NA's :1   
## pm25\_mia summer\_tmmx summer\_rmax winter\_tmmx   
## Min. :0.00000 Min. :290.5 Min. :31.64 Min. :264.7   
## 1st Qu.:0.00000 1st Qu.:300.8 1st Qu.:88.05 1st Qu.:275.1   
## Median :0.00000 Median :303.3 Median :91.32 Median :280.2   
## Mean :0.00354 Mean :303.1 Mean :88.97 Mean :280.4   
## 3rd Qu.:0.00000 3rd Qu.:305.8 3rd Qu.:94.81 3rd Qu.:285.5   
## Max. :1.00000 Max. :313.9 Max. :99.78 Max. :298.3   
##   
## winter\_rmax bmcruderate   
## Min. :58.16 Min. : 189.3   
## 1st Qu.:85.09 1st Qu.: 864.3   
## Median :88.03 Median :1036.3   
## Mean :87.47 Mean :1029.2   
## 3rd Qu.:90.75 3rd Qu.:1194.1   
## Max. :97.67 Max. :1978.6   
##

apply(Covid\_df, 2, sd, na.rm = TRUE)

## Warning in var(if (is.vector(x) || is.factor(x)) x else as.double(x), na.rm =  
## na.rm): NAs introduced by coercion

## state deathspc intersects\_msa   
## NA 6.785215e+01 4.906356e-01   
## cur\_smoke\_q1 cur\_smoke\_q2 cur\_smoke\_q3   
## 1.493481e-01 1.281304e-01 1.321812e-01   
## cur\_smoke\_q4 bmi\_obese\_q1 bmi\_obese\_q2   
## 1.101103e-01 1.659285e-01 1.532368e-01   
## bmi\_obese\_q3 bmi\_obese\_q4 exercise\_any\_q1   
## 1.758494e-01 1.672267e-01 2.738741e-01   
## exercise\_any\_q2 exercise\_any\_q3 exercise\_any\_q4   
## 3.223363e-01 3.578608e-01 3.769216e-01   
## brfss\_mia puninsured2010 reimb\_penroll\_adj10   
## 4.327567e-01 5.536651e+00 1.590926e+03   
## mort\_30day\_hosp\_z adjmortmeas\_amiall30day adjmortmeas\_chfall30day   
## 1.206493e+00 3.940837e-02 2.356548e-02   
## med\_prev\_qual\_z primcarevis\_10 diab\_hemotest\_10   
## 8.638807e-01 7.401457e+00 6.594153e+00   
## diab\_eyeexam\_10 diab\_lipids\_10 mammogram\_10   
## 7.598549e+00 7.854145e+00 8.397699e+00   
## cs00\_seg\_inc cs00\_seg\_inc\_pov25 cs00\_seg\_inc\_aff75   
## 3.057628e-02 3.075727e-02 3.292040e-02   
## cs\_race\_theil\_2000 gini99 poor\_share   
## 8.413111e-02 8.667691e-02 6.545970e-02   
## inc\_share\_1perc frac\_middleclass scap\_ski90pcm   
## 5.063134e-02 9.309948e-02 1.347960e+00   
## rel\_tot cs\_frac\_black cs\_frac\_hisp   
## 1.850252e+01 1.448372e+01 1.205040e+01   
## unemp\_rate cs\_labforce cs\_elf\_ind\_man   
## 1.773790e-02 7.039307e-02 9.086221e-02   
## cs\_born\_foreign mig\_inflow mig\_outflow   
## 4.836270e+00 1.903371e-02 1.378019e-02   
## pop\_density frac\_traveltime\_lt15 hhinc00   
## 1.676096e+03 1.372145e-01 6.975837e+03   
## median\_house\_value ccd\_exp\_tot score\_r   
## 6.318905e+04 2.103573e+00 9.007980e+00   
## cs\_fam\_wkidsinglemom subcty\_exp\_pc taxrate   
## 6.782804e-02 9.998335e+02 1.384751e-02   
## tax\_st\_diff\_top20 pm25 pm25\_mia   
## 1.470989e+00 2.565927e+00 5.940534e-02   
## summer\_tmmx summer\_rmax winter\_tmmx   
## 3.173951e+00 9.689271e+00 6.597855e+00   
## winter\_rmax bmcruderate   
## 4.811207e+00 2.483818e+02

# Question 3:

Note that some variables have missing values. This causes problems when estimating the models. Normally we’d impute missing values by replacing them with their mean or median value, but to keep things simple, given the size of our data, you should drop all observations (rows) with missing values.

Covid\_df <- na.omit(Covid\_df)

# Question 4:

Create a separate dummy variable for each of the 48 states and the District of Columbia in the dataset (so you’ll create 49 dummy variables in total).

states <- unique(Covid\_df$state)  
  
dummy\_states <- sapply(states, function(x) as.numeric(Covid\_df$state == x))  
  
colnames(dummy\_states) <- states  
  
Covid\_df <- cbind(Covid\_df, dummy\_states)  
  
Covid\_df$state <- NULL  
  
Covid\_df$county <- NULL

# Question 5:

Split the sample into training (80% of the data) and test (20% of the data) sets. Be sure to set a seed so you can replicate your work.

set.seed(123)  
  
n\_obs <- nrow(Covid\_df)  
  
split\_index <- sample(seq\_len(n\_obs),  
 size = floor(0.8 \* n\_obs),  
 replace = FALSE  
)  
  
train\_data <- Covid\_df[split\_index, ]  
  
test\_data <- Covid\_df[-split\_index, ]

# Question 6

Using the training data, estimate the relationship between COVID-19 deaths per capita (y = deathspc) and the Opportunity Insights and PM COVID predictors listed in the spreadsheet, as well as state-level fixed effects (the state dummy variables) using OLS.

# Part A:

Based on those estimates, calculate and report the MSE and R2 in both the training and test sets.

OLS\_model\_train <- lm(deathspc ~ ., data = train\_data)  
  
Train\_prediction <- predict(OLS\_model\_train, newdata = test\_data)

## Warning in predict.lm(OLS\_model\_train, newdata = test\_data): prediction from a  
## rank-deficient fit may be misleading

MSE\_Train <- mean((test\_data$deathspc - Train\_prediction)^2)  
  
output\_1 <- paste("The MSE for the Training Data", MSE\_Train)  
  
print(output\_1)

## [1] "The MSE for the Training Data 1589.29241323927"

train\_r2 <- 1 - MSE\_Train / var(train\_data$deathspc)  
  
train\_r2

## [1] 0.4346467

summary(OLS\_model\_train)

##   
## Call:  
## lm(formula = deathspc ~ ., data = train\_data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -171.18 -16.13 -4.89 6.66 575.15   
##   
## Coefficients: (2 not defined because of singularities)  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 8.285e+01 2.362e+02 0.351 0.725836   
## intersects\_msa 1.803e+00 2.434e+00 0.740 0.459076   
## cur\_smoke\_q1 -9.603e-01 9.762e+00 -0.098 0.921653   
## cur\_smoke\_q2 -3.979e+00 9.812e+00 -0.406 0.685117   
## cur\_smoke\_q3 -1.650e+00 7.899e+00 -0.209 0.834559   
## cur\_smoke\_q4 4.634e+00 9.262e+00 0.500 0.616861   
## bmi\_obese\_q1 -6.549e+00 9.331e+00 -0.702 0.482848   
## bmi\_obese\_q2 5.910e+00 9.245e+00 0.639 0.522745   
## bmi\_obese\_q3 -1.361e+01 6.656e+00 -2.045 0.040997 \*   
## bmi\_obese\_q4 -1.883e+00 6.740e+00 -0.279 0.779994   
## exercise\_any\_q1 -6.936e+00 8.311e+00 -0.835 0.404011   
## exercise\_any\_q2 1.756e+01 8.205e+00 2.140 0.032480 \*   
## exercise\_any\_q3 -1.953e+00 6.879e+00 -0.284 0.776569   
## exercise\_any\_q4 -2.514e+00 7.464e+00 -0.337 0.736236   
## brfss\_mia -5.528e-01 8.559e+00 -0.065 0.948513   
## puninsured2010 -7.011e-01 4.562e-01 -1.537 0.124449   
## reimb\_penroll\_adj10 -1.269e-03 1.054e-03 -1.203 0.228936   
## mort\_30day\_hosp\_z 1.466e+00 2.055e+00 0.713 0.475716   
## adjmortmeas\_amiall30day -2.032e+01 4.759e+01 -0.427 0.669402   
## adjmortmeas\_chfall30day 1.506e+01 7.902e+01 0.191 0.848876   
## med\_prev\_qual\_z 8.752e+00 5.430e+00 1.612 0.107138   
## primcarevis\_10 -3.094e-01 1.981e-01 -1.562 0.118540   
## diab\_hemotest\_10 -1.014e+00 2.889e-01 -3.511 0.000456 \*\*\*  
## diab\_eyeexam\_10 -9.248e-02 2.477e-01 -0.373 0.708965   
## diab\_lipids\_10 -2.283e-01 2.539e-01 -0.899 0.368518   
## mammogram\_10 -2.599e-01 2.245e-01 -1.158 0.246942   
## cs00\_seg\_inc 1.302e+03 4.955e+02 2.629 0.008631 \*\*   
## cs00\_seg\_inc\_pov25 -8.462e+02 2.611e+02 -3.241 0.001211 \*\*   
## cs00\_seg\_inc\_aff75 -5.217e+02 2.523e+02 -2.068 0.038785 \*   
## cs\_race\_theil\_2000 1.309e+01 1.536e+01 0.852 0.394196   
## gini99 -3.896e+01 2.733e+01 -1.426 0.154084   
## poor\_share -1.654e+00 3.983e+01 -0.042 0.966880   
## inc\_share\_1perc -2.858e+00 3.425e+01 -0.083 0.933501   
## frac\_middleclass -6.995e+01 2.277e+01 -3.072 0.002152 \*\*   
## scap\_ski90pcm -5.109e+00 1.449e+00 -3.526 0.000430 \*\*\*  
## rel\_tot 1.453e-01 8.095e-02 1.795 0.072806 .   
## cs\_frac\_black 7.869e-01 1.649e-01 4.772 1.94e-06 \*\*\*  
## cs\_frac\_hisp -3.714e-02 1.631e-01 -0.228 0.819908   
## unemp\_rate -1.824e+02 8.576e+01 -2.127 0.033536 \*   
## cs\_labforce -5.657e+01 2.849e+01 -1.986 0.047183 \*   
## cs\_elf\_ind\_man 3.556e+01 1.602e+01 2.220 0.026527 \*   
## cs\_born\_foreign 1.357e+00 4.069e-01 3.335 0.000867 \*\*\*  
## mig\_inflow -2.024e+01 1.100e+02 -0.184 0.854070   
## mig\_outflow -2.464e+02 1.476e+02 -1.669 0.095297 .   
## pop\_density 9.793e-03 7.006e-04 13.977 < 2e-16 \*\*\*  
## frac\_traveltime\_lt15 -1.226e+01 1.332e+01 -0.921 0.357322   
## hhinc00 7.630e-04 3.852e-04 1.981 0.047713 \*   
## median\_house\_value -7.727e-06 3.422e-05 -0.226 0.821368   
## ccd\_exp\_tot 1.460e+00 7.025e-01 2.078 0.037779 \*   
## score\_r 2.253e-01 1.741e-01 1.294 0.195646   
## cs\_fam\_wkidsinglemom 1.622e+01 3.820e+01 0.425 0.671209   
## subcty\_exp\_pc -9.642e-04 1.247e-03 -0.773 0.439521   
## taxrate -5.663e+01 1.231e+02 -0.460 0.645595   
## tax\_st\_diff\_top20 4.347e+01 5.431e+01 0.800 0.423508   
## pm25 -4.063e-01 1.034e+00 -0.393 0.694267   
## pm25\_mia 5.855e+00 2.461e+01 0.238 0.811968   
## summer\_tmmx 1.754e-01 9.537e-01 0.184 0.854077   
## summer\_rmax -2.974e-01 3.506e-01 -0.848 0.396277   
## winter\_tmmx 6.531e-01 7.130e-01 0.916 0.359797   
## winter\_rmax -5.724e-01 4.110e-01 -1.393 0.163849   
## bmcruderate -3.748e-05 8.352e-03 -0.004 0.996421   
## Alabama -6.936e+00 1.478e+01 -0.469 0.638918   
## Arizona -9.399e+01 8.805e+01 -1.067 0.285864   
## Arkansas -5.720e+01 4.636e+01 -1.234 0.217376   
## California -3.076e+02 3.372e+02 -0.912 0.361739   
## Colorado 2.359e+01 1.166e+01 2.023 0.043235 \*   
## Connecticut 9.915e+01 2.248e+01 4.411 1.08e-05 \*\*\*  
## Delaware -1.106e+00 4.281e+01 -0.026 0.979386   
## Florida -1.259e+01 1.563e+01 -0.805 0.420641   
## Georgia 1.610e+01 1.434e+01 1.122 0.261774   
## Idaho -1.393e+01 2.026e+01 -0.688 0.491805   
## Illinois 1.424e+01 1.317e+01 1.081 0.279805   
## Indiana 4.650e+01 1.341e+01 3.468 0.000534 \*\*\*  
## Iowa -1.057e+02 1.446e+02 -0.731 0.464767   
## Kansas 1.062e+00 8.783e+00 0.121 0.903739   
## Kentucky 3.394e+00 9.426e+00 0.360 0.718801   
## Louisiana -9.664e+01 2.092e+02 -0.462 0.644245   
## Maine 2.574e+00 1.836e+01 0.140 0.888520   
## Maryland -4.232e+01 4.573e+01 -0.925 0.354930   
## Massachusetts 8.683e+01 2.018e+01 4.302 1.76e-05 \*\*\*  
## Michigan 3.780e+01 1.316e+01 2.872 0.004114 \*\*   
## Minnesota -9.329e+01 1.245e+02 -0.750 0.453587   
## Mississippi 1.250e+01 1.443e+01 0.866 0.386497   
## Missouri 6.245e+00 1.278e+01 0.489 0.625201   
## Montana 2.071e+01 1.253e+01 1.652 0.098599 .   
## Nebraska -5.848e+01 8.446e+01 -0.692 0.488704   
## Nevada -1.373e+01 1.721e+01 -0.798 0.425148   
## `New Hampshire` 1.055e+01 1.918e+01 0.550 0.582550   
## `New Mexico` -1.496e+01 1.443e+01 -1.037 0.299856   
## `New York` 5.055e+01 1.380e+01 3.663 0.000255 \*\*\*  
## `North Carolina` -5.577e+01 4.589e+01 -1.215 0.224370   
## `North Dakota` -1.290e+02 1.760e+02 -0.733 0.463766   
## Ohio -9.408e+01 1.345e+02 -0.699 0.484395   
## Oklahoma 5.152e+00 1.336e+01 0.386 0.699736   
## Oregon 5.921e+00 1.345e+01 0.440 0.659832   
## Pennsylvania 2.102e+01 1.375e+01 1.529 0.126393   
## `Rhode Island` -2.725e+02 3.250e+02 -0.839 0.401836   
## `South Carolina` -2.321e+01 1.558e+01 -1.489 0.136534   
## `South Dakota` 1.994e+01 1.364e+01 1.463 0.143688   
## Tennessee -8.117e-02 1.397e+01 -0.006 0.995364   
## Texas -1.173e+01 1.369e+01 -0.857 0.391369   
## Utah -6.362e+00 1.504e+01 -0.423 0.672255   
## Vermont -2.554e+02 3.097e+02 -0.824 0.409790   
## Virginia -1.491e+01 1.360e+01 -1.096 0.273074   
## Washington 1.787e+01 1.342e+01 1.331 0.183349   
## `West Virginia` -1.006e+02 1.261e+02 -0.798 0.425131   
## Wisconsin NA NA NA NA   
## Wyoming NA NA NA NA   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 41.22 on 2226 degrees of freedom  
## Multiple R-squared: 0.4227, Adjusted R-squared: 0.3955   
## F-statistic: 15.52 on 105 and 2226 DF, p-value: < 2.2e-16

OLS\_model\_test <- lm(deathspc ~ ., data = test\_data)  
  
Test\_prediction <- predict(OLS\_model\_test, newdata = train\_data)

## Warning in predict.lm(OLS\_model\_test, newdata = train\_data): prediction from a  
## rank-deficient fit may be misleading

MSE\_Test <- mean((train\_data$deathspc - Test\_prediction)^2)  
  
output\_2 <- paste("The MSE for the Test Data", MSE\_Test)  
  
print(output\_2)

## [1] "The MSE for the Test Data 2176.75559766276"

test\_r2 <- 1 - MSE\_Test / var(test\_data$deathspc)  
  
test\_r2

## [1] 0.08504075

summary(OLS\_model\_test)

##   
## Call:  
## lm(formula = deathspc ~ ., data = test\_data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -101.244 -18.845 -3.507 11.401 258.585   
##   
## Coefficients: (3 not defined because of singularities)  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -6.535e+02 5.147e+02 -1.270 0.204831   
## intersects\_msa 4.904e+00 4.949e+00 0.991 0.322224   
## cur\_smoke\_q1 -1.134e-01 2.228e+01 -0.005 0.995942   
## cur\_smoke\_q2 -1.460e+01 2.409e+01 -0.606 0.544741   
## cur\_smoke\_q3 7.828e+00 1.875e+01 0.418 0.676446   
## cur\_smoke\_q4 -1.663e+01 1.782e+01 -0.933 0.351212   
## bmi\_obese\_q1 8.176e+00 1.572e+01 0.520 0.603117   
## bmi\_obese\_q2 7.954e+00 2.158e+01 0.369 0.712637   
## bmi\_obese\_q3 -2.525e+01 1.378e+01 -1.833 0.067395 .   
## bmi\_obese\_q4 2.721e+01 1.400e+01 1.943 0.052552 .   
## exercise\_any\_q1 4.769e+00 1.540e+01 0.310 0.756873   
## exercise\_any\_q2 1.920e+01 1.807e+01 1.063 0.288480   
## exercise\_any\_q3 -5.667e+00 1.760e+01 -0.322 0.747587   
## exercise\_any\_q4 -2.306e+01 1.251e+01 -1.844 0.065812 .   
## brfss\_mia -1.725e+01 2.113e+01 -0.816 0.414691   
## puninsured2010 2.755e-01 9.912e-01 0.278 0.781196   
## reimb\_penroll\_adj10 -1.063e-03 2.228e-03 -0.477 0.633540   
## mort\_30day\_hosp\_z -2.826e+00 4.469e+00 -0.632 0.527468   
## adjmortmeas\_amiall30day 7.238e+01 1.069e+02 0.677 0.498704   
## adjmortmeas\_chfall30day -3.803e+01 1.773e+02 -0.214 0.830275   
## med\_prev\_qual\_z -6.200e-01 1.053e+01 -0.059 0.953069   
## primcarevis\_10 2.557e-01 4.062e-01 0.630 0.529261   
## diab\_hemotest\_10 -4.832e-01 6.623e-01 -0.730 0.466011   
## diab\_eyeexam\_10 1.101e-02 5.453e-01 0.020 0.983899   
## diab\_lipids\_10 -2.287e-01 5.460e-01 -0.419 0.675513   
## mammogram\_10 4.749e-01 4.723e-01 1.005 0.315235   
## cs00\_seg\_inc -8.661e+01 9.376e+02 -0.092 0.926441   
## cs00\_seg\_inc\_pov25 -3.352e+02 4.932e+02 -0.680 0.497131   
## cs00\_seg\_inc\_aff75 4.551e+02 4.757e+02 0.957 0.339196   
## cs\_race\_theil\_2000 -2.449e+01 3.035e+01 -0.807 0.419993   
## gini99 -3.275e+01 6.057e+01 -0.541 0.588931   
## poor\_share 2.231e+02 8.294e+01 2.690 0.007390 \*\*   
## inc\_share\_1perc 2.205e+01 7.189e+01 0.307 0.759236   
## frac\_middleclass -4.317e+01 4.641e+01 -0.930 0.352758   
## scap\_ski90pcm -1.597e+00 2.911e+00 -0.549 0.583441   
## rel\_tot 1.121e-01 1.557e-01 0.720 0.471831   
## cs\_frac\_black 9.702e-01 3.369e-01 2.880 0.004152 \*\*   
## cs\_frac\_hisp 2.402e-03 4.216e-01 0.006 0.995457   
## unemp\_rate -1.471e+02 1.662e+02 -0.885 0.376473   
## cs\_labforce 1.017e+01 5.885e+01 0.173 0.862839   
## cs\_elf\_ind\_man 8.885e+01 3.355e+01 2.648 0.008357 \*\*   
## cs\_born\_foreign -9.943e-01 1.123e+00 -0.886 0.376208   
## mig\_inflow -1.284e+02 2.213e+02 -0.580 0.562052   
## mig\_outflow -2.173e+01 3.035e+02 -0.072 0.942959   
## pop\_density 3.729e-03 3.943e-03 0.946 0.344773   
## frac\_traveltime\_lt15 -1.892e+01 2.565e+01 -0.737 0.461268   
## hhinc00 1.322e-03 8.157e-04 1.621 0.105767   
## median\_house\_value 1.075e-04 1.161e-04 0.927 0.354649   
## ccd\_exp\_tot -8.155e-01 2.259e+00 -0.361 0.718243   
## score\_r 1.605e-01 3.787e-01 0.424 0.671879   
## cs\_fam\_wkidsinglemom -6.885e+01 7.760e+01 -0.887 0.375383   
## subcty\_exp\_pc 6.480e-03 3.108e-03 2.085 0.037632 \*   
## taxrate 1.833e+02 2.739e+02 0.669 0.503684   
## tax\_st\_diff\_top20 -3.566e+00 1.027e+01 -0.347 0.728660   
## pm25 3.401e-02 2.030e+00 0.017 0.986639   
## pm25\_mia 1.180e+01 3.418e+01 0.345 0.730166   
## summer\_tmmx 3.279e-01 2.022e+00 0.162 0.871252   
## summer\_rmax -1.662e+00 7.811e-01 -2.128 0.033855 \*   
## winter\_tmmx 2.076e+00 1.465e+00 1.416 0.157281   
## winter\_rmax 1.141e+00 8.390e-01 1.360 0.174478   
## bmcruderate -2.740e-04 1.746e-02 -0.016 0.987486   
## Alabama -3.480e+01 2.756e+01 -1.263 0.207230   
## Arizona -6.416e+01 5.321e+01 -1.206 0.228444   
## Arkansas -3.977e+01 2.077e+01 -1.915 0.056106 .   
## California -5.255e+01 5.770e+01 -0.911 0.362886   
## Colorado -3.181e+01 3.110e+01 -1.023 0.306817   
## Connecticut 1.035e+02 2.846e+01 3.636 0.000306 \*\*\*  
## Delaware 4.122e+01 4.272e+01 0.965 0.335118   
## Florida -4.520e+01 3.324e+01 -1.360 0.174559   
## Georgia 1.509e+01 2.758e+01 0.547 0.584596   
## Idaho -1.750e+01 2.811e+01 -0.623 0.533906   
## Illinois 1.567e+01 1.890e+01 0.829 0.407540   
## Indiana 1.970e+01 2.011e+01 0.980 0.327733   
## Iowa 2.818e+01 2.635e+01 1.069 0.285493   
## Kansas -1.850e+01 2.188e+01 -0.846 0.398228   
## Kentucky 1.528e+00 2.063e+01 0.074 0.941013   
## Louisiana 2.994e+01 3.299e+01 0.908 0.364577   
## Maine 1.990e+01 2.273e+01 0.876 0.381671   
## Maryland 1.981e+01 2.333e+01 0.849 0.396261   
## Massachusetts 1.222e+02 2.492e+01 4.906 1.28e-06 \*\*\*  
## Michigan 1.773e+01 1.641e+01 1.081 0.280335   
## Minnesota 1.801e+01 2.546e+01 0.707 0.479758   
## Mississippi -2.916e+01 2.921e+01 -0.998 0.318637   
## Missouri -1.397e+01 2.107e+01 -0.663 0.507528   
## Montana -2.649e+01 2.524e+01 -1.049 0.294514   
## Nebraska 4.995e+00 1.928e+01 0.259 0.795673   
## Nevada -6.329e+01 4.578e+01 -1.383 0.167448   
## `New Hampshire` -1.050e+01 3.265e+01 -0.322 0.747920   
## `New Mexico` -1.636e+01 3.535e+01 -0.463 0.643756   
## `New York` 1.276e+01 1.861e+01 0.686 0.493055   
## `North Carolina` -2.673e+01 1.932e+01 -1.383 0.167282   
## `North Dakota` 3.318e+01 3.413e+01 0.972 0.331424   
## Ohio 8.121e+00 2.243e+01 0.362 0.717423   
## Oklahoma -1.800e+01 2.677e+01 -0.673 0.501546   
## Oregon -2.808e+01 3.351e+01 -0.838 0.402450   
## Pennsylvania 2.753e+01 1.909e+01 1.442 0.149898   
## `Rhode Island` 2.425e+01 5.837e+01 0.415 0.677980   
## `South Carolina` -4.789e+01 3.012e+01 -1.590 0.112415   
## `South Dakota` -5.162e+00 2.089e+01 -0.247 0.804958   
## Tennessee -2.176e+01 2.268e+01 -0.959 0.337817   
## Texas -3.780e+01 2.924e+01 -1.293 0.196690   
## Utah -8.722e+01 4.767e+01 -1.830 0.067942 .   
## Vermont 4.738e+01 5.744e+01 0.825 0.409896   
## Virginia -2.714e+01 2.326e+01 -1.167 0.243873   
## Washington -2.706e+01 3.110e+01 -0.870 0.384820   
## `West Virginia` NA NA NA NA   
## Wisconsin NA NA NA NA   
## Wyoming NA NA NA NA   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 39.25 on 478 degrees of freedom  
## Multiple R-squared: 0.4683, Adjusted R-squared: 0.3526   
## F-statistic: 4.047 on 104 and 478 DF, p-value: < 2.2e-16

The MSE for the Training data is equal to 1589.29241323927

The R^2 for the Training data is equal to 0.4346467

The MSE for the Test data is equal to 2176.75559766276

The R^2 for the Test data is equal to 0.08504075

MSE Difference = 587

# Part B:

Is there any evidence of overfitting? Briefly explain

Yes, there is evidence of overfitting when the training model performs significantly better than the test model. The mean squared error (MSE) for the training model was much lower than that of the test model, indicating that the model is fitting the noise in the training data too closely instead of the underlying pattern. This leads to high variance and low bias, resulting in a failure to generalize well to new data. Ultimately, this creates a model that performs well on the training data but poorly on new, unseen data. We also know that the MSE continues to decrease as the model becomes more complex (adding more predicting variables). However, the MSE for the test data will decrease initially, but then increase overtime as we continue to add more co-variates.

# Question 7

Use the training set to estimate Ridge Regression and the Lasso analogs to the OLS model in the previous question. For each, you should report a plot of the cross-validation estimates of the test error as a function of the value of the hyperparameter (λ) that indicates the tuned value of λ. Hint: to do so you should be sure standardize your predictors and tune the hyperparameter by:

1. Calculating each model for a grid or range of values of λ. You’ll want to adjust the values you use based on the data, but start by using 100 values of λ from 0.01 to 100.
2. Using 10-fold cross-validation (10FCV) (on the training set) to estimate the test error for each model at the given value of λ.
3. Plotting the cross-validation estimates of the test error as a function of the value of λ.
4. Choosing the optimal value of λ.
5. Re-estimating your model using that optimal value of λ

# Question 7 Ridge Regression

# Part A:

library(glmnet)  
  
set.seed(321)  
  
# Standardize predictors  
  
stndrd\_ridge1 <- model.matrix(OLS\_model\_train)  
  
# Set up grid of lambda values  
  
a1 <- seq(-2, 2, by = 1/25)  
  
r1 <- 10^a1  
  
# Fit Ridge Regression model for each value of lambda  
  
ridge\_model1 <- glmnet(stndrd\_ridge1,  
 train\_data$deathspc,  
 alpha = 0,  
 lambda = r1  
)

# Part B:

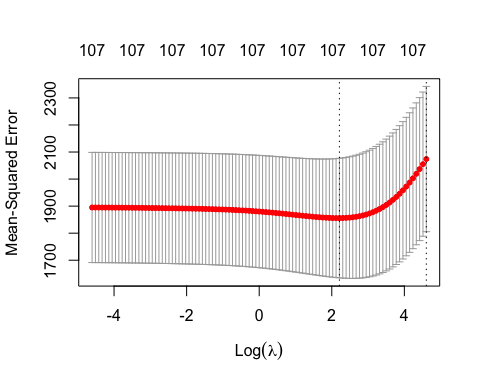
Perform 10-fold cross-validation to estimate test error

cv\_ridge1 <- cv.glmnet(stndrd\_ridge1,  
 train\_data$deathspc,  
 alpha = 0,  
 lambda = r1,  
 nfolds = 10  
)

# Part C:

Plot cross-validation estimates of test error

plot(cv\_ridge1)



# Part D:

Choose optimal value of lambda

opt\_lambda1 <- cv\_ridge1$lambda.min  
  
output\_3 <- paste("The optimal value of lambda for the ridge regression is", opt\_lambda1)  
  
print(output\_3)

## [1] "The optimal value of lambda for the ridge regression is 9.1201083935591"

# Part: E

Re-estimate model using optimal value of lambda

ridge\_model\_opt1 <- glmnet(stndrd\_ridge1,  
 train\_data$deathspc,  
 alpha = 0,  
 lambda = opt\_lambda1  
)

# Question 7 Lasso Regression:

# Part A:

library(glmnet)  
  
set.seed(321)  
  
# Standardize predictors  
  
stndrd\_lasso2 <- model.matrix(OLS\_model\_train)  
  
# Set up grid of lambda values  
  
a2 <- seq(-2, 2, by = 1 / 25)  
  
l2 <- 10^a2  
  
# Fit Lasso Regression model for each value of lambda  
  
lasso\_model2 <- glmnet(stndrd\_lasso2,  
 train\_data$deathspc,  
 alpha = 1,  
 lambda = l2  
)

# Part B:

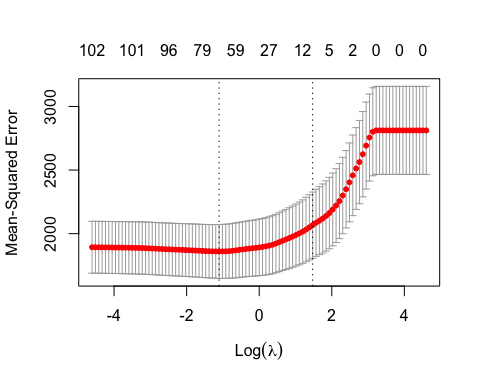
Perform 10-fold cross-validation to estimate test error

cv\_lasso2 <- cv.glmnet(stndrd\_lasso2,  
 train\_data$deathspc,  
 alpha = 1,  
 lambda = l2,  
 nfolds = 10  
)

# Part C:

Plot cross-validation estimates of test error

plot(cv\_lasso2)



# Part D:

Choose optimal value of lambda

lambda\_opt2 <- cv\_lasso2$lambda.min  
  
output\_4 <- paste("The optimal value of lambda for the lasso regression is", lambda\_opt2)  
  
print(output\_4)

## [1] "The optimal value of lambda for the lasso regression is 0.331131121482591"

# Part E:

Re-estimate model using optimal value of lambda

lasso\_model\_opt2 <- glmnet(stndrd\_lasso2,  
 train\_data$deathspc,  
 alpha = 1,  
 lambda = lambda\_opt2  
)

# Question 8:

Using the optimal values of λ you found for Ridge Regression and the Lasso in the previous question, calculate and report the training- and test-set prediction errors (MSE & R2) for each model. Did Ridge Regression and/or the Lasso mitigate overfitting? Briefly explain your results.

# Ridge Regression Training Set

stndrd\_ridge3 <- model.matrix(OLS\_model\_test)  
  
ridge\_train\_pred <- predict(ridge\_model\_opt1, newx = stndrd\_ridge3)  
  
ridge\_train\_mse <- mean((test\_data$deathspc - ridge\_train\_pred)^2)  
  
ridge\_train\_mse

## [1] 1603.056

ridge\_train\_r2 <- 1 - ridge\_train\_mse / var(train\_data$deathspc)  
  
ridge\_train\_r2

## [1] 0.4297505

# Ridge Regression Test Set

stndrd\_ridge3 <- model.matrix(OLS\_model\_test)  
  
# Set up grid of lambda values  
  
a3 <- seq(-2, 2, by = 1 / 25)  
  
r3 <- 10^a1  
  
# Fit Ridge Regression model for each value of lambda  
  
ridge\_model3 <- glmnet(stndrd\_ridge3,  
 test\_data$deathspc,  
 alpha = 0,  
 lambda = r3  
)  
  
cv\_ridge3 <- cv.glmnet(stndrd\_ridge3,  
 test\_data$deathspc,  
 alpha = 0,  
 lambda = r3,  
 nfolds = 10  
)  
  
opt\_lambda3 <- cv\_ridge3$lambda.min  
  
ridge\_model\_opt3 <- glmnet(stndrd\_ridge3,  
 test\_data$deathspc,  
 alpha = 0,  
 lambda = opt\_lambda3  
)  
  
ridge\_test\_pred <- predict(ridge\_model\_opt3, newx = stndrd\_ridge1)  
  
ridge\_test\_mse <- mean((train\_data$deathspc - ridge\_test\_pred)^2)  
  
ridge\_test\_mse

## [1] 2093.761

ridge\_test\_r2 <- 1 - ridge\_test\_mse / var(test\_data$deathspc)  
  
ridge\_test\_r2

## [1] 0.119926

The MSE for the Ridge Regression Training data is equal to 1603.056

The R^2 for the Ridge Regression Training data is equal to 0.4297505

The MSE for the Ridge Regression Test data is equal to 2093.761

The R^2 for the Ridge Regression Test data is equal to 0.119926

MSE Difference = 490

# 

# Lasso Regression Training Set

stndrd\_lasso4 <- model.matrix(OLS\_model\_test)  
  
lasso\_train\_pred <- predict(lasso\_model\_opt2,  
 newx = stndrd\_lasso4  
)  
  
lasso\_train\_mse <- mean((test\_data$deathspc - lasso\_train\_pred)^2)  
  
lasso\_train\_mse

## [1] 1594.639

lasso\_train\_r2 <- 1 - lasso\_train\_mse / var(train\_data$deathspc)  
  
lasso\_train\_r2

## [1] 0.4327447

# Lasso Regression Test Set

stndrd\_lasso4 <- model.matrix(OLS\_model\_test)  
  
a4 <- seq(-2, 2, by = 1 / 25)  
  
l4 <- 10^a2  
  
# Fit Lasso Regression model for each value of lambda  
  
lasso\_model4 <- glmnet(stndrd\_lasso4,  
 test\_data$deathspc,  
 alpha = 1,  
 lambda = l4  
)  
  
cv\_lasso4 <- cv.glmnet(stndrd\_lasso4,  
 test\_data$deathspc,  
 alpha = 1,  
 lambda = l4,  
 nfolds = 10  
)  
  
lambda\_opt4 <- cv\_lasso4$lambda.min  
  
lasso\_model\_opt4 <- glmnet(stndrd\_lasso4,  
 test\_data$deathspc,  
 alpha = 1,  
 lambda = lambda\_opt4  
)  
  
lasso\_test\_pred <- predict(lasso\_model\_opt4,  
 newx = stndrd\_lasso2  
)  
  
lasso\_test\_mse <- mean((train\_data$deathspc - lasso\_test\_pred)^2)  
  
lasso\_test\_mse

## [1] 2154.324

lasso\_test\_r2 <- 1 - lasso\_test\_mse / var(test\_data$deathspc)  
  
lasso\_test\_r2

## [1] 0.09446952

The MSE for the Lasso Regression Training data is equal to 1594.639

The R^2 for the Lasso Regression Training data is equal to 0.4327447

The MSE for the Lasso Regression Test data is equal to 2154.324

The R^2 for the Lasso Regression Test data is equal to 0.09446952

MSE Difference = 560

Both Lasso and Ridge Regression are effective in reducing overfitting by minimizing the difference between the mean squared error (MSE) of the training and test datasets, compared to the original Ordinary Least Squares (OLS) model. However, Ridge Regression is more effective than Lasso in reducing the MSE difference to a greater extent. Therefore, in this example, Ridge Regression is generally considered to be the preferred method for reducing overfitting.