ISYE-6501
Fall 2024

Homework 7

Question 11.1
Using the crime data set uscrime.txt from Questions 8.2, 9.1, and 10.1, build a regression model using

1. **Stepwise regression**
2. **Lasso**
3. **Elastic net**

For Parts 2 and 3, remember to scale the data first - otherwise, the regression coefficients will be on different scales and the constraint won't have the desired effect.

ANALYSIS:
I ran the three different regression models. The stepwise was able to give an adj r-squared of 0.7444. The lasso was able to give us an adj r-squared of 0.7292 and then increase to 0.7307 once I removed the coefficients that had a p-value greater than 0.05. The elastic net was able to provide an ajd r-squared of 0.714. While the stepwise regression could have been overfitting, I would say that based on the adjusted r-squared values that were obtained, the best model of fit was with the stepwise regression. The next best model of fit was the lasso after we were able to calculate the alpha value with the highest r2 which ended up being an alpha of 0.5. Lastly was the elastic net with the lowest adj r-squared value of 0.714.

Below is the code/ R markdown, find the input in black, the comments in green, and the output in blue.

CODE:
```
> #analyze the current data
> datamodel <-lm(Crime~., data = data)
> summary(datamodel)
Call:
lm(formula = Crime ~ ., data = data)

Residuals:
    Min      1Q  Median      3Q     Max 
-395.74  -98.09   -6.69  112.99  512.67 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) -5.984e+03  1.628e+03  -3.675 0.000893 ***
M            8.783e+01  4.171e+01   2.106 0.043443 *  
So          -3.803e+00  1.488e+02  -0.026 0.979765    
Ed           1.883e+02  6.209e+01   3.033 0.004861 ** 
Po1          1.928e+02  1.061e+02   1.817 0.078892 .  
Po2         -1.094e+02  1.175e+02  -0.931 0.358830    
```

```
LF        -6.638e+02  1.470e+03  -0.452 0.654654
M.F        1.741e+01  2.035e+01   0.855 0.398995
Pop       -7.330e-01  1.290e+00  -0.568 0.573845
NW         4.204e+00  6.481e+00   0.649 0.521279
U1        -5.827e+03  4.210e+03  -1.384 0.176238
U2         1.678e+02  8.234e+01   2.038 0.050161 .
Wealth     9.617e-02  1.037e-01   0.928 0.360754
Ineq       7.067e+01  2.272e+01   3.111 0.003983 **
Prob      -4.855e+03  2.272e+03  -2.137 0.040627 *
Time      -3.479e+00  7.165e+00  -0.486 0.630708
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 209.1 on 31 degrees of freedom
Multiple R-squared:  0.8031,	Adjusted R-squared:  0.7078
F-statistic: 8.429 on 15 and 31 DF,  p-value: 3.539e-07

> #perform a stepwise regression
> stepwisemodel <- train(Crime ~., data = data, method = "lmStepAIC", trControl = trainControl(), trace = FALSE)
> #analyze the stepwise model
> summary(stepwisemodel$finalModel)

Call:
lm(formula = .outcome ~ M + Ed + Po1 + M.F + U1 + U2 + Ineq +
    Prob, data = dat)

Residuals:
    Min      1Q  Median      3Q     Max
-444.70 -111.07    3.03  122.15  483.30

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -6426.10    1194.61  -5.379 4.04e-06 ***
M              93.32      33.50   2.786  0.00828 **
Ed            180.12      52.75   3.414  0.00153 **
Po1           102.65      15.52   6.613 8.26e-08 ***
M.F            22.34      13.60   1.642  0.10874
U1          -6086.63    3339.27  -1.823  0.07622 .
U2            187.35      72.48   2.585  0.01371 *
Ineq           61.33      13.96   4.394 8.63e-05 ***
Prob        -3796.03    1490.65  -2.547  0.01505 *
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 195.5 on 38 degrees of freedom
Multiple R-squared:  0.7888,	Adjusted R-squared:  0.7444 
F-statistic: 17.74 on 8 and 38 DF,  p-value: 1.159e-10

> #Now for the lasso method we have to start by scaling the data.
> datascale <- cbind(as.data.frame(scale(data[,1])), as.data.frame(data[,2]),
as.data.frame(scale(data[,c(3,4,5,6,7,8,9,10,11,12,13,14,15)])), as.data.frame(data[,16]))
> #Now we replace teh column names
> colnames(datascale) <- colnames(data)
> #Check to see if data is scaled
> summary(datascale)
       M                 So               Ed               Po1               Po2         
 Min.   :-1.5575   Min.   :0.0000   Min.   :-1.6661   Min.   :-1.3459   Min.   :-1.4032  
 1st Qu.:-0.6823   1st Qu.:0.0000   1st Qu.:-0.7275   1st Qu.:-0.7571   1st Qu.:-0.7773  
 Median :-0.2048   Median :0.0000   Median : 0.2111   Median :-0.2355   Median :-0.2587  
 Mean   : 0.0000   Mean   :0.3404   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000  
 3rd Qu.: 0.5908   3rd Qu.:1.0000   3rd Qu.: 0.7921   3rd Qu.: 0.6561   3rd Qu.: 0.5996  
 Max.   : 3.0575   Max.   :1.0000   Max.   : 1.4626   Max.   : 2.7255   Max.   : 2.7454  
       LF                M.F               Pop                NW                U1         
 Min.   :-2.00910   Min.   :-1.6636   Min.   :-0.8830   Min.   :-0.9640   Min.   :-1.4126  
 1st Qu.:-0.75947   1st Qu.:-0.6285   1st Qu.:-0.6991   1st Qu.:-0.7501   1st Qu.:-0.8302  
 Median :-0.02948   Median :-0.2043   Median :-0.3051   Median :-0.2444   Median :-0.1924  
 Mean   : 0.00000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000  
 3rd Qu.: 0.78711   3rd Qu.: 0.3047   3rd Qu.: 0.1283   3rd Qu.: 0.3051   3rd Qu.: 0.4732  
 Max.   : 1.97488   Max.   : 2.9856   Max.   : 3.4510   Max.   : 3.1302   Max.   : 2.5810  
       U2                Wealth             Ineq               Prob              Time        
 Min.   :-1.655178   Min.   :-2.4602   Min.   :-1.7044   Min.   :-1.7677   Min.   :-2.0317  
 1st Qu.:-0.767126   1st Qu.:-0.6828   1st Qu.:-0.7144   1st Qu.:-0.6329   1st Qu.:-0.7052  
 Median : 0.002519   Median : 0.1204   Median :-0.4512   Median :-0.2195   Median :-0.1125  
 Mean   : 0.000000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000  
 3rd Qu.: 0.535351   3rd Qu.: 0.6852   3rd Qu.: 0.8397   3rd Qu.: 0.3236   3rd Qu.: 0.5437  
 Max.   : 2.844286   Max.   : 1.6957   Max.   : 2.0553   Max.   : 3.1980   Max.   : 2.4556  
     Crime       
 Min.   : 342.0  
 1st Qu.: 658.5  
 Median : 831.0  
 Mean   : 905.1  
 3rd Qu.:1057.5  
 Max.   :1993.0  
```

```
> #Now we run the lasso
> datalasso <- cv.glmnet(x=as.matrix(datascale[,-16]), y=as.matrix(datascale$Crime), alpha=1, nfolds = 5,
type.measure="mse", family="gaussian")
> #Now we look for the smallest cvm lambda
> x <- datalasso$cvm
> which(x == min(x))
[1] 36
> datalasso$lambda.min
[1] 10.13846
> coefficients(datalasso, s=datalasso$lambda.min)
16 x 1 sparse Matrix of class "dgCMatrix"
                   s1
(Intercept) 889.3103622
M            86.9317163
So           46.3383111
Ed          131.7765650
Po1         307.7067455
Po2           .
LF            0.1168486
M.F          54.0530574
Pop           .
NW            5.1858569
U1          -29.9110342
U2           64.4278417
Wealth        .
Ineq        185.1622530
Prob        -83.0905697
Time          .
> #Now we make a regression model with the right coefficients
> datalassolm <- lm(Crime ~M+So+Ed+Po1+M.F+NW+U1+U2+Wealth+Ineq+Prob, data = datascale)
> #and assess the Rsquared values
> summary(datalassolm)

Call:
lm(formula = Crime ~ M + So + Ed + Po1 + M.F + NW + U1 + U2 + 
    Wealth + Ineq + Prob, data = datascale)

Residuals:
    Min      1Q  Median      3Q     Max 
-408.38  -96.14   -1.39  114.80  454.53 

Coefficients:
```

```
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   893.73      51.33  17.411  < 2e-16 ***
M             114.97      48.92   2.350  0.02454 *
So             33.35     123.69   0.270  0.78905
Ed            195.31      62.52   3.124  0.00357 **
Po1           275.69      59.99   4.596 5.41e-05 ***
M.F            64.50      42.82   1.506  0.14101
NW             15.93      57.16   0.279  0.78209
U1            -94.61      64.90  -1.458  0.15380
U2            140.81      66.32   2.123  0.04089 *
Wealth         73.59      93.96   0.783  0.43878
Ineq          267.01      80.66   3.310  0.00217 **
Prob          -87.64      40.25  -2.177  0.03627 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 201.3 on 35 degrees of freedom
Multiple R-squared:  0.794,	Adjusted R-squared:  0.7292
F-statistic: 12.26 on 11 and 35 DF,  p-value: 5.334e-09

> #Now lets only try using the coefficients with p-value less than .05
> datalassolm2 <- lm(Crime ~M+Ed+Po1+U2+Ineq+Prob, data = datascale)
> summary(datalassolm2)

Call:
lm(formula = Crime ~ M + Ed + Po1 + U2 + Ineq + Prob, data = datascale)

Residuals:
    Min      1Q  Median      3Q     Max
-470.68  -78.41  -19.68  133.12  556.23

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   905.09      29.27  30.918  < 2e-16 ***
M             131.98      41.85   3.154  0.00305 **
Ed            219.79      50.07   4.390 8.07e-05 ***
Po1           341.84      40.87   8.363 2.56e-10 ***
U2             75.47      34.55   2.185  0.03483 *
Ineq          269.91      55.60   4.855 1.88e-05 ***
Prob          -86.44      34.74  -2.488  0.01711 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 200.7 on 40 degrees of freedom
Multiple R-squared:  0.7659,	Adjusted R-squared:  0.7307
F-statistic: 21.81 on 6 and 40 DF,  p-value: 3.418e-11

> #Now we run the elasticnet regression trying different alpha values 0-1
> r2=c()
> for (i in 0:10) {
+   dataelasticmodel <- cv.glmnet(x=as.matrix(datascale[,-16]),y=as.matrix(datascale$Crime),
+                                 alpha=i/10, nfolds = 5, type.measure="mse",
+                                 family="gaussian")
+   r2 = cbind(r2, dataelasticmodel$glmnet.fit$dev.ratio[which(dataelasticmodel$glmnet.fit$lambda == dataelasticmodel$lambda.min)])
+ }
> #Now we find the alpha with best r2.
> alpha <- (which.max(r2)-1)/10
> alpha
[1] 0.5
> dataelastic <- cv.glmnet(x=as.matrix(datascale[,-16]), y=as.matrix(datascale$Crime), alpha=0.05, nfolds = 5, type.measure="mse", family="gaussian")
> dataelasticm = lm(Crime ~M+So+Ed+Po1+Po2+LF+M.F+NW+U1+U2+Wealth+Ineq+Prob+Time, data = datascale)
> summary(dataelasticm)

Call:
lm(formula = Crime ~ M + So + Ed + Po1 + Po2 + LF + M.F + NW + 
    U1 + U2 + Wealth + Ineq + Prob + Time, data = datascale)

Residuals:
    Min      1Q  Median      3Q     Max 
-380.91 -101.89  -14.77  110.87  505.40 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  906.483     58.484  15.500  < 2e-16 ***
M            112.837     51.691   2.183  0.03649 *  
So            -4.105    147.172  -0.028  0.97792    
Ed           211.246     68.713   3.074  0.00429 ** 
Po1          563.337    311.541   1.808  0.07998 .  
Po2         -313.824    324.701  -0.966  0.34104    
LF           -31.702     58.147  -0.545  0.58939    
M.F           64.479     54.722   1.178  0.24737    
```

```
NW          44.572     65.892   0.676  0.50362
U1        -112.728     73.902  -1.525  0.13699
U2         143.186     68.749   2.083  0.04535 *
Wealth      87.836     98.588   0.891  0.37961
Ineq       269.086     86.824   3.099  0.00403 **
Prob      -110.457     51.117  -2.161  0.03830 *
Time       -31.582     48.772  -0.648  0.52189
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 206.8 on 32 degrees of freedom
Multiple R-squared:  0.801,	Adjusted R-squared:  0.714
F-statistic: 9.202 on 14 and 32 DF,  p-value: 1.301e-07
```