Question 11.1

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

- **Stepwise regression**
- **Lasso**
- **Elastic net**

Set my work directory, cleared the environment and set a seed.

```
setwd("C:/Users/ryand/OneDrive/Desktop/MSAO/Datasets")
rm(list = ls())
set.seed(42)
```

Read in the crime dataset; aggregated statistics for 47 states from the 60's.

```
dataload=read.csv("uscrime.csv",sep="")
dataload
head(dataload)
tail(dataload)
```

I used the randomForest function and 'importance' results to help prioritize the predictors in the Crime dataset, which helped me decide what order I would add factors when building a regression model using the stepwise method.

```
library(randomForest)
b<-randomForest(Crime~.,dataload, importance=TRUE, ntree=500, mtry=5)
b$importance
```

	%IncMSE	IncNodePurity
M	2351.8348	228719.50
So	686.3600	23456.11
Ed	3789.9004	213283.48
Po1	33343.2293	1095484.85
Po2	31013.2581	1197840.76
LF	3417.6595	270248.43
M.F	1482.1069	232785.09
Pop	996.0957	343523.34
NW	16991.3555	520590.05
U1	-1583.5395	133339.11
U2	1764.8321	162803.16
Wealth	4669.1996	654055.84
Ineq	1589.2677	208959.89
Prob	17467.4152	743181.12
Time	607.2216	231615.22

Using the stepwise method, I added factors one at a time and ran the regression model using the lm() function with each additional factor. I used the jtools package and summ() function to scale the predictors and resulting coefficients of the model. Each predictor that had a p-value less than or equal to 0.015 was kept in the model, and those that were greater than 0.15 were removed. For the final model, I required all predictors to have a p-value of less than or equal to 0.10 in order to remain in the model. Here are the coefficients and R squared of my final model:

MODEL INFO:

Observations: 47

Dependent Variable: Crime

Type: OLS linear regression

MODEL FIT:

F(3,43) = 16.65, p = 0.00

R^2 = 0.54

Adj. R^2 = 0.51

Standard errors: OLS

	Est.	S.E.	t val.	p
(Intercept)	905.09	39.69	22.80	0.00
Po1	277.81	41.12	6.76	0.00
NW	96.64	43.43	2.23	0.03
LF	72.27	42.74	1.69	0.10

This model explains 51% of the variability that exists in the Crime rate between the 47 states.

Using the glmnet() function, I ran a Lasso regression model on the same Crime dataset; the best model used all 15 of the predictors and explained 80.3% of the variability of Crime rate. Here are the regression coefficients of the **Lasso** model:

library(glmnet)

x <- as.matrix(dataload[,-16])

y <-as.vector(dataload[,16])

#elastic-lasso

model1<-glmnet(x,y,standardize = TRUE,alpha=1)

print(model1)

coef(model1,s=0.080)

```
                    s1
(Intercept) -6.015970e+03
M            8.785250e+01
So          -1.027400e-01
Ed           1.865095e+02
Po1          1.822653e+02
Po2         -9.740560e+01
LF          -5.998614e+02
M.F          1.741086e+01
Pop         -7.300763e-01
NW           3.950903e+00
U1          -5.723468e+03
U2           1.670495e+02
Wealth       9.473198e-02
Ineq         7.053662e+01
Prob        -4.784177e+03
Time        -3.174739e+00
```

Using the glmnet() function, I ran a Ridge regression model on the same Crime dataset; the best model used all 15 of the predictors and explained 77.43% of the variability of Crime rate. Here are the regression coefficients of the **Ridge** model:

#elastic-ridge

model2<-glmnet(x,y,standardize = TRUE,alpha=0)

print(model2)

coef(model2,s=26)

```
                   s1
(Intercept) -5.579394e+03
M            7.065651e+01
So           7.950176e+01
Ed           1.149492e+02
Po1          5.508492e+01
Po2          3.772236e+01
LF           4.177259e+02
M.F          2.280885e+01
Pop         -1.826204e-01
NW           2.824573e+00
U1          -3.335806e+03
U2           1.188814e+02
Wealth       4.545313e-02
Ineq         4.394975e+01
Prob        -4.029724e+03
Time         2.821382e-01
```

Appendix – R Code

```
setwd("C:/Users/ryand/OneDrive/Desktop/MSAO/Datasets")
#clear global environment
rm(list = ls())
#setting a seed which allows for reproducible results; sets the same sequence of randomly generated #'s
set.seed(1)
```

```
#reading in the data set
dataload=read.csv("uscrime.csv",sep="")
dataload
#checking data
head(dataload)
tail(dataload)
#####random forest, using random forest's "importance" ranking as the order in which I will add factors to the stepwise regression.
library(randomForest)
#mtry sets the # of random splits the random tree function will make with each iteration (good rule to use is n/3 where n is the # of predictors.)
b<-randomForest(Crime~.,dataload, importance=TRUE, ntree=500, mtry=5)
b$importance

#Calling the jtools package to use it's scaling ability on the model coefficients
library(jtools)
#performing a manual stepwise regression method.
fit1<-lm(Crime~Po1,data=dataload)
summ(fit1,scale = TRUE)

fit2<-lm(Crime~Po1+Po2,data=dataload)
summ(fit2,scale = TRUE)

fit3<-lm(Crime~Po1+Prob,data=dataload)
summ(fit3,scale = TRUE)

fit4<-lm(Crime~Po1+NW, data=dataload)
summ(fit4,scale = TRUE)

fit5<-lm(Crime~Po1+NW+Wealth, data=dataload)
summ(fit5,scale = TRUE)

fit6<-lm(Crime~Po1+NW+Ed, data=dataload)
summ(fit6,scale = TRUE)

###the fit7 linear regression model is the final model, has enough factors (at least 10 points per factor), and all are significant with a pvalue less than 0.05
fit7<-lm(Crime~Po1+NW+LF, data=dataload)
summ(fit7,scale = TRUE)

fit8<-lm(Crime~Po1+NW+LF+M, data=dataload)
summ(fit8,scale = TRUE)

fit9<-lm(Crime~Po1+NW+LF+U2, data=dataload)
summ(fit9,scale = TRUE)
```

```
fit10<-lm(Crime~Po1+NW+LF+Ineq, data=dataload)
summ(fit10,scale = TRUE)

fit11<-lm(Crime~Po1+NW+LF+U1, data=dataload)
summ(fit11,scale = TRUE)

fit12<-lm(Crime~Po1+NW+LF+M.F, data=dataload)
summ(fit12,scale = TRUE)

fit13<-lm(Crime~Po1+NW+LF+Pop, data=dataload)
summ(fit13,scale = TRUE)

fit14<-lm(Crime~Po1+NW+LF+So, data=dataload)
summ(fit14,scale = TRUE)

fit15<-lm(Crime~Po1+NW+LF+Time, data=dataload)
summ(fit15,scale = TRUE)

library(glmnet)
#https://rstudio-pubs-static.s3.amazonaws.com/482594_1ad8bd60595d4f9a8f65c3a57a8ce96a.html
#https://glmnet.stanford.edu/articles/glmnet.html#:~:text=Glmnet%20is%20a%20package%20that,for%20the%20regul
arization%20parameter%20lambda.
x <- as.matrix(dataload[,-16])
y <-as.vector(dataload[,16])

#elastic-lasso
model1<-glmnet(x,y,standardize = TRUE,alpha=1)
print(model1)
coef(model1,s=0.080)
model1ssres<-sum((as.data.frame(predict(model1, newx = x, type = "response", s = 0.080))-(dataload[,16]))^2)
model1sstot<-sum((mean(dataload[,16])-(dataload[,16]))^2)
model1rsq<-1-(model1ssres/model1sstot)
model1rsq

#elastic-ridge
model2<-glmnet(x,y,standardize = TRUE,alpha=0)
print(model2)
coef(model2,s=26)
model2ssres<-sum((as.data.frame(predict(model2, newx = x, type = "response", s = 26))-(dataload[,16]))^2)
model2sstot<-sum((mean(dataload[,16])-(dataload[,16]))^2)
model2rsq<-1-(model2ssres/model2sstot)
model2rsq
```