LOOCV & LASSO Regression

Alex Weirth

2023-02-23

Load Libraries

```
library(tidyverse)
library(mgcv)
library(magrittr)
```

Import Data

```
life_df <- read.csv("life_df.csv")
```

```
life_df <- na.omit(life_df)
```

Testing & Training Data

```
dim(life_df) # 1649
```

```
## [1] 1649   24
```

```
set.seed(123)

#70-30 Split
trainInd<-sample(1:1649, 1155)

life_df_train<-life_df[trainInd, ]
life_df_test<-life_df[-trainInd, ]
```

```
colnames(life_df)
```

```
##  [1] "country"            "year"               "status"
##  [4] "life_exp_yrs"       "adult_mortality"    "infant_deaths"
##  [7] "alcohol"            "perc_expend"        "hep_b"
## [10] "measles"            "bmi"                "X5yr_deaths"
## [13] "polio"              "tot_expend"         "diphtheria"
## [16] "hiv_aids"           "gdp"                "population"
## [19] "thin_1to19"         "thin_5to9"          "inc_comp_resources"
## [22] "schooling"          "hiv_deaths_cat"     "bmi_cat"
```

```
dim(life_df)
```

```
## [1] 1649   24
```

```
colnames(life_df)
```

```
##  [1] "country"            "year"               "status"
```

```
##  [4] "life_exp_yrs"       "adult_mortality"    "infant_deaths"
##  [7] "alcohol"            "perc_expend"        "hep_b"
## [10] "measles"            "bmi"                "X5yr_deaths"
## [13] "polio"              "tot_expend"         "diphtheria"
## [16] "hiv_aids"           "gdp"                "population"
## [19] "thin_1to19"         "thin_5to9"          "inc_comp_resources"
## [22] "schooling"          "hiv_deaths_cat"     "bmi_cat"
```

```
#life_df <- life_df %>%
  #rename("five_yr_deaths" = "X5yr_deaths")
```

```
str(life_df_train)
```

```
## 'data.frame':    1155 obs. of  24 variables:
##  $ country           : chr  "Cyprus" "Ecuador" "Benin" "Fiji" ...
##  $ year              : int  2013 2003 2014 2000 2011 2003 2007 2004 2009 2008 ...
##  $ status            : chr  "Developed" "Developing" "Developing" "Developing" ...
##  $ life_exp_yrs      : num  81 74.4 59.7 67.7 68.3 72.7 76.4 73 68.2 68 ...
##  $ adult_mortality   : int  54 151 252 221 225 112 124 134 261 282 ...
##  $ infant_deaths     : int  0 8 25 0 0 0 1 1 15 18 ...
##  $ alcohol           : num  9.04 3.69 0.01 2.05 0.23 ...
##  $ perc_expend       : num  212.1 18.3 90.1 31.3 289.9 ...
##  $ hep_b             : int  96 82 78 98 95 98 85 89 98 97 ...
##  $ measles           : int  0 0 786 0 10 75 0 11 101 6 ...
##  $ bmi               : num  59.2 45.8 25.2 5.2 21.1 16.7 51.9 53.3 57 17.1 ...
##  $ X5yr_deaths       : int  0 10 39 0 1 0 2 1 18 28 ...
##  $ polio             : int  99 97 74 91 95 98 84 96 98 97 ...
##  $ tot_expend        : num  7.46 6.46 4.59 3.87 4.73 5.9 6.31 8.24 7.44 7.66 ...
##  $ diphtheria        : int  99 87 78 9 95 98 85 88 98 97 ...
##  $ hiv_aids          : num  0.1 0.3 1.1 0.1 0.5 0.1 0.2 0.1 0.3 3.7 ...
##  $ gdp               : num  2798 244 944 276 2458 ...
##  $ population        : num  1143896 1328961 1286712 811223 7451 ...
##  $ thin_1to19        : num  0.9 1.5 7.1 4.3 16.3 14.6 2.1 2.6 2.3 6.7 ...
##  $ thin_5to9         : num  1 1.4 6.9 4 17 14.7 2 2.6 2.5 6.6 ...
##  $ inc_comp_resources: num  0.85 0.679 0.475 0.681 0.572 0.601 0.743 0.72 0.776 0.438 ...
##  $ schooling         : num  13.8 12.6 10.7 13.1 11.9 11.8 12.9 13.1 14 10.4 ...
##  $ hiv_deaths_cat    : chr  "Under 1 Death/1000" "Under 1 Death/1000" "Over 1 Death/1000" "Under 1 De
##  $ bmi_cat           : chr  "obese" "obese" "overweight" "underweight" ...
##  - attr(*, "na.action")= 'omit' Named int [1:1289] 33 45 46 47 48 49 58 59 60 61 ...
##   ..- attr(*, "names")= chr [1:1289] "33" "45" "46" "47" ...
```

Part I: Use LOOCV (Leave-One-Out Cross Validation) to perform best subsets and find the best number of variables to use.

```
### best subset model
library(leaps)
life_bestsub.model <- regsubsets(life_exp_yrs ~
                            adult_mortality +
                            infant_deaths +
                            alcohol +
                            perc_expend +
                            hep_b +
```

```
                              measles +
                              bmi +
                              X5yr_deaths +
                              polio +
                              tot_expend +
                              diphtheria +
                              hiv_aids +
                              gdp +
                              population +
                              thin_1to19 +
                              thin_5to9 +
                              inc_comp_resources +
                              schooling,
                            data = life_df_train,
                            nvmax = 18)

summary(life_bestsub.model)
```

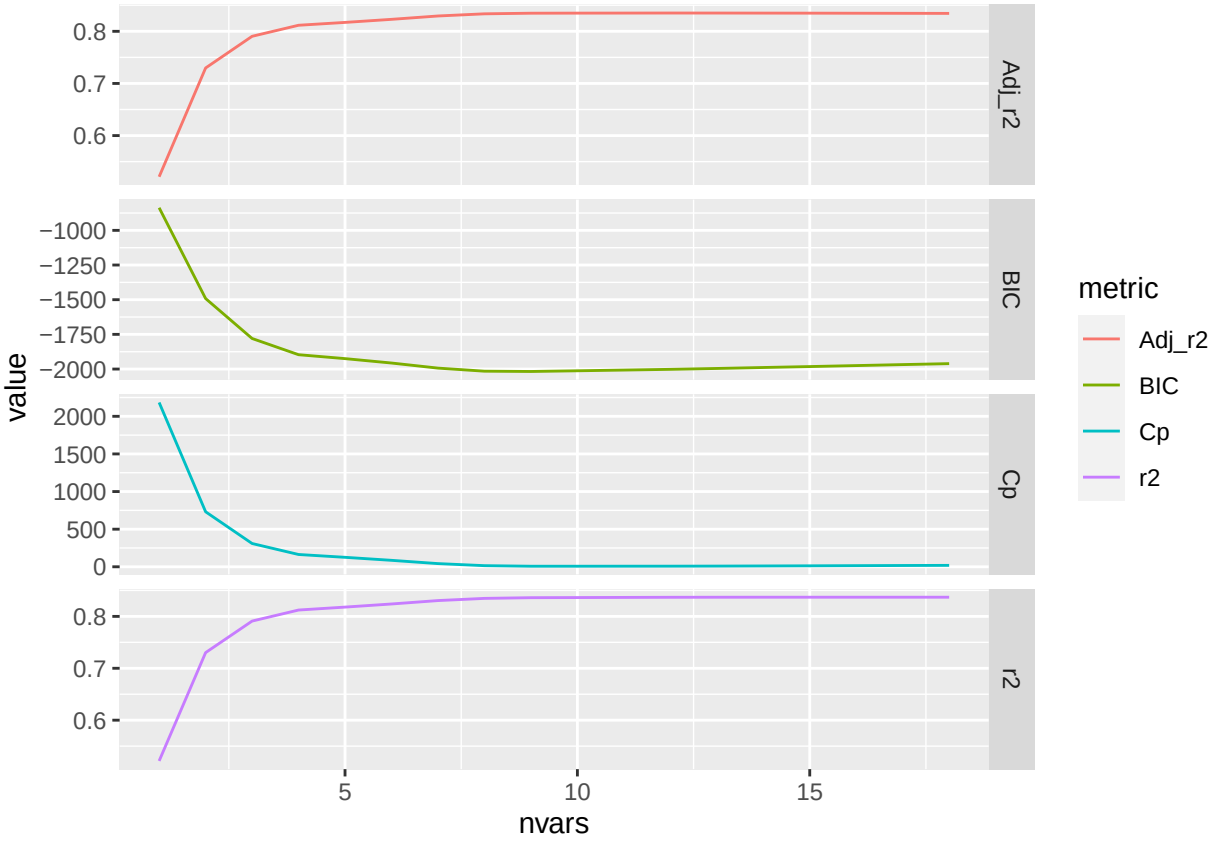
```
## Subset selection object
## Call: regsubsets.formula(life_exp_yrs ~ adult_mortality + infant_deaths +
##     alcohol + perc_expend + hep_b + measles + bmi + X5yr_deaths +
##     polio + tot_expend + diphtheria + hiv_aids + gdp + population +
##     thin_1to19 + thin_5to9 + inc_comp_resources + schooling,
##     data = life_df_train, nvmax = 18)
## 18 Variables  (and intercept)
##                     Forced in Forced out
## adult_mortality         FALSE      FALSE
## infant_deaths           FALSE      FALSE
## alcohol                 FALSE      FALSE
## perc_expend             FALSE      FALSE
## hep_b                   FALSE      FALSE
## measles                 FALSE      FALSE
## bmi                     FALSE      FALSE
## X5yr_deaths             FALSE      FALSE
## polio                   FALSE      FALSE
## tot_expend              FALSE      FALSE
## diphtheria              FALSE      FALSE
## hiv_aids                FALSE      FALSE
## gdp                     FALSE      FALSE
## population              FALSE      FALSE
## thin_1to19              FALSE      FALSE
## thin_5to9               FALSE      FALSE
## inc_comp_resources      FALSE      FALSE
## schooling               FALSE      FALSE
## 1 subsets of each size up to 18
## Selection Algorithm: exhaustive
##          adult_mortality infant_deaths alcohol perc_expend hep_b measles bmi
## 1  ( 1 )  " "             " "           " "     " "         " "   " "     " "
## 2  ( 1 )  " "             " "           " "     " "         " "   " "     " "
## 3  ( 1 )  "*"             " "           " "     " "         " "   " "     " "
## 4  ( 1 )  "*"             " "           " "     " "         " "   " "     " "
## 5  ( 1 )  "*"             " "           " "     "*"         " "   " "     " "
## 6  ( 1 )  "*"             "*"           " "     " "         " "   " "     " "
## 7  ( 1 )  "*"             "*"           " "     "*"         " "   " "     " "
```

```
## 8  ( 1 )  "*"             "*"           " "     "*"         " "   " "     "*"
## 9  ( 1 )  "*"             "*"           " "     "*"         " "   " "     "*"
## 10  ( 1 ) "*"             "*"           " "     "*"         " "   " "     "*"
## 11  ( 1 ) "*"             "*"           "*"     "*"         " "   " "     "*"
## 12  ( 1 ) "*"             "*"           "*"     "*"         " "   " "     "*"
## 13  ( 1 ) "*"             "*"           "*"     "*"         " "   " "     "*"
## 14  ( 1 ) "*"             "*"           "*"     "*"         " "   " "     "*"
## 15  ( 1 ) "*"             "*"           "*"     "*"         " "   "*"     "*"
## 16  ( 1 ) "*"             "*"           "*"     "*"         "*"   "*"     "*"
## 17  ( 1 ) "*"             "*"           "*"     "*"         "*"   "*"     "*"
## 18  ( 1 ) "*"             "*"           "*"     "*"         "*"   "*"     "*"
##           X5yr_deaths polio tot_expend diphtheria hiv_aids gdp population
## 1  ( 1 )  " "         " "   " "        " "         " "      " " " "
## 2  ( 1 )  " "         " "   " "        " "         "*"      " " " "
## 3  ( 1 )  " "         " "   " "        " "         "*"      " " " "
## 4  ( 1 )  " "         " "   " "        " "         "*"      " " " "
## 5  ( 1 )  " "         " "   " "        " "         "*"      " " " "
## 6  ( 1 )  "*"         " "   " "        " "         "*"      " " " "
## 7  ( 1 )  "*"         " "   " "        " "         "*"      " " " "
## 8  ( 1 )  "*"         " "   " "        " "         "*"      " " " "
## 9  ( 1 )  "*"         "*"   " "        " "         "*"      " " " "
## 10  ( 1 ) "*"         "*"   "*"        " "         "*"      " " " "
## 11  ( 1 ) "*"         "*"   "*"        " "         "*"      " " " "
## 12  ( 1 ) "*"         "*"   "*"        " "         "*"      " " " "
## 13  ( 1 ) "*"         "*"   "*"        "*"         "*"      " " " "
## 14  ( 1 ) "*"         "*"   "*"        "*"         "*"      "*" " "
## 15  ( 1 ) "*"         "*"   "*"        "*"         "*"      "*" " "
## 16  ( 1 ) "*"         "*"   "*"        "*"         "*"      "*" " "
## 17  ( 1 ) "*"         "*"   "*"        "*"         "*"      "*" "*"
## 18  ( 1 ) "*"         "*"   "*"        "*"         "*"      "*" "*"
##           thin_1to19 thin_5to9 inc_comp_resources schooling
## 1  ( 1 )  " "        " "       " "                "*"
## 2  ( 1 )  " "        " "       " "                "*"
## 3  ( 1 )  " "        " "       " "                "*"
## 4  ( 1 )  " "        " "       "*"                "*"
## 5  ( 1 )  " "        " "       "*"                "*"
## 6  ( 1 )  " "        " "       "*"                "*"
## 7  ( 1 )  " "        " "       "*"                "*"
## 8  ( 1 )  " "        " "       "*"                "*"
## 9  ( 1 )  " "        " "       "*"                "*"
## 10  ( 1 ) " "        " "       "*"                "*"
## 11  ( 1 ) " "        " "       "*"                "*"
## 12  ( 1 ) " "        "*"       "*"                "*"
## 13  ( 1 ) " "        "*"       "*"                "*"
## 14  ( 1 ) " "        "*"       "*"                "*"
## 15  ( 1 ) " "        "*"       "*"                "*"
## 16  ( 1 ) " "        "*"       "*"                "*"
## 17  ( 1 ) " "        "*"       "*"                "*"
## 18  ( 1 ) "*"        "*"       "*"                "*"
```

Model Metrics

```
#performance measures
best18<-data.frame(nvars=1:18,
```

```
  Cp     = summary(life_bestsub.model)$cp,
  r2     = summary(life_bestsub.model)$rsq,
  Adj_r2 = summary(life_bestsub.model)$adjr2,
  BIC    =summary(life_bestsub.model)$bic)%>%
  gather(metric, value, -c(nvars))

ggplot(best18, aes(x=nvars, y=value, color=metric))+
  geom_line()+
  facet_grid(metric~., scales = "free")
```




```
which.max(summary(life_bestsub.model)$adjr2)
```

Maximizing adj_r2

```
## [1] 12
```

```
which.max(summary(life_bestsub.model)$bic)
```

Minimizing BIC

```
## [1] 1
```

```
which.max(summary(life_bestsub.model)$cp)
```

Minimizing Cp

```
## [1] 1
```

```
which.max(summary(life_bestsub.model)$rsq)
```

Maximizing r2

```
## [1] 18
```

Cross Validation (LOOCV)

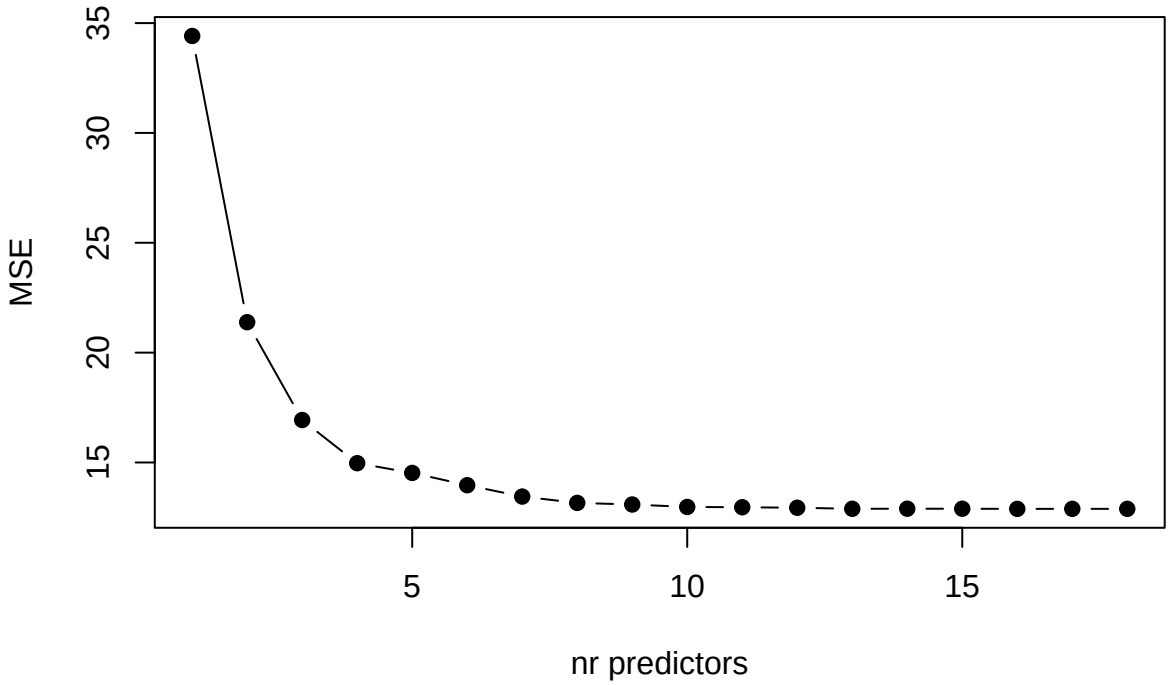
```
life_df_train <- na.omit(life_df_train)
life_df_test <- na.omit(life_df_test)
dim(life_df_train)
```

```
## [1] 1155   24
```

```
##jack-knife validation (leave-one-out)

##Function to get predictions from the regsubset function
predict.regsubsets <- function(object, newdata, id,...){
  form  <- as.formula(object$call[[2]])
  mat   <- model.matrix(form, newdata)
  coefi <- coef(object, id=id)
  mat[, names(coefi)]%*%coefi
}

#store the prediction error n=252
jk.errors <- matrix(NA, 1164, 18)

for (k in 1:1164){
  #uses regsubsets in the data with 1 observation removed
  best.model.cv <- regsubsets(life_exp_yrs ~
                              adult_mortality +
                              infant_deaths +
                              alcohol +
                              perc_expend +
                              hep_b +
                              measles +
                              bmi +
                              X5yr_deaths +
                              polio +
                              tot_expend +
                              diphtheria +
                              hiv_aids +
                              gdp +
                              population +
                              thin_1to19 +
                              thin_5to9 +
                              inc_comp_resources +
                              schooling,
                            data = life_df_train[-k,],
                            nvmax = 18)

  #Models with 18 predictors
```

```
  for (i in 1:18){
    #that was left out
    pred <- predict.regsubsets(best.model.cv,                #prediction in the obsv
                    life_df[k,],
                    id=i)
    jk.errors[k,i] <- (life_df$life_exp_yrs[k]-pred)^2       #error in the obsv
  }
}

mse.models <- apply(jk.errors, 2, mean)
#MSE estimation
```

```
plot(mse.models,                                  #Plot with MSEs
     pch=19, type="b",
     xlab="nr predictors",
     ylab="MSE")
```



Final Model

The best final model has 9 variables which minimizes the MSE.

```
loocv_finalmod <- lm(life_exp_yrs ~ adult_mortality +
                       infant_deaths + perc_expend +
                       X5yr_deaths + hiv_aids +
                       inc_comp_resources +
                       schooling + hiv_deaths_cat,
                       data = life_df_train)
```

Multicollinearity in the LASSO model?

```
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
```

```
vif(lm(life_exp_yrs ~ adult_mortality +
                      infant_deaths +
                      perc_expend +
                      X5yr_deaths +
                      hiv_aids +
                      inc_comp_resources +
                      schooling +
                      hiv_deaths_cat,
                      data = life_df_train))
```

```
##    adult_mortality      infant_deaths        perc_expend        X5yr_deaths
##           1.845612         269.022273           1.232172         269.279007
##           hiv_aids inc_comp_resources          schooling     hiv_deaths_cat
##           1.697275           2.647895           2.881842           2.125358
```

There are features with a VIF factor over 10 which means they definitely have strong multicollinearity. A regularization method will protect against this...

Part II: Perform LASSO regression

```
## LASSO Model
library(glmnet)
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

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

```
## Loaded glmnet 4.1-6
```

```
library(faraway)
```

```
##
## Attaching package: 'faraway'
```

```
## The following objects are masked from 'package:car':
##
##     logit, vif
```

```
set.seed(3)
```

```r
#LASSO REGRESSION

dim(life_df)
```

```
## [1] 1649   24
```

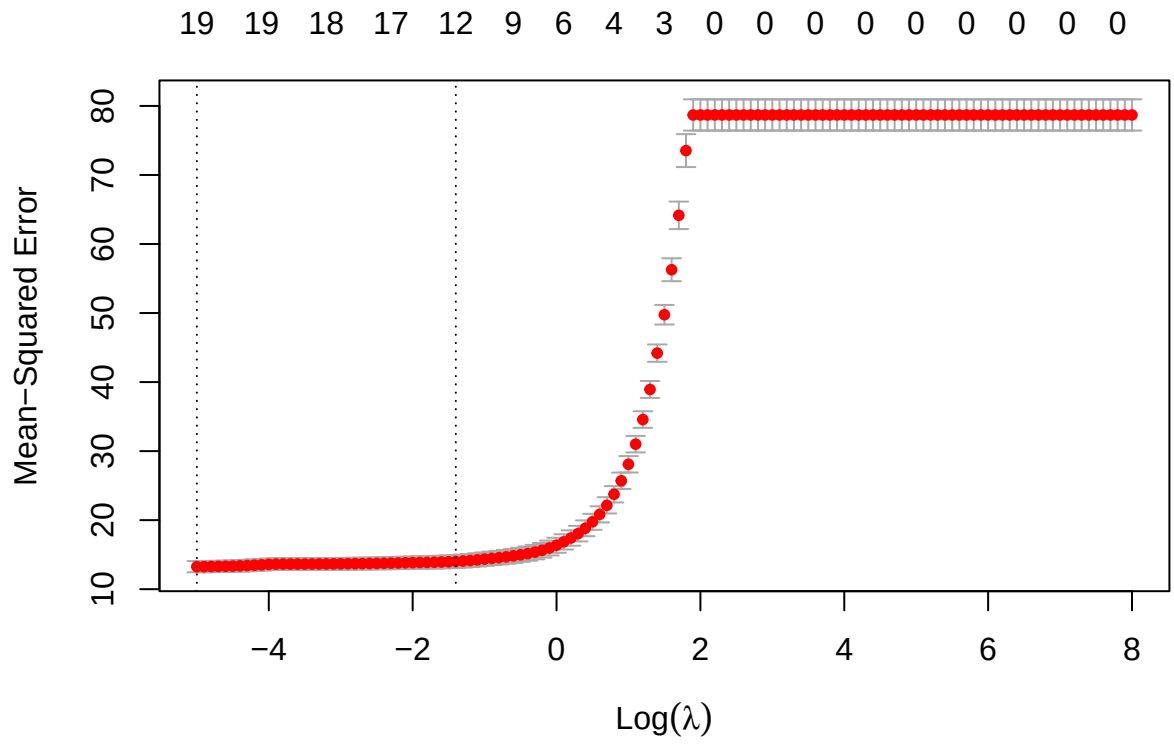
```r
# Defining the model equation

# Took out country because it wasn't doing anything and was using countries name to predict life exp
X <- model.matrix(life_exp_yrs ~ year +
                    status + adult_mortality +
                    infant_deaths + alcohol +
                    perc_expend + hep_b + measles +
                    bmi + X5yr_deaths +
                    polio + tot_expend +
                    diphtheria + hiv_aids +
                    gdp + population +
                    thin_1to19 + thin_5to9+
                    inc_comp_resources + schooling, data = life_df_train)[,-1]


# Defining the outcome
Y <- life_df_train$life_exp_yrs
length(Y)
```

```
## [1] 1155
```

```r
head(Y)
```

```
## [1] 81.0 74.4 59.7 67.7 68.3 72.7
```

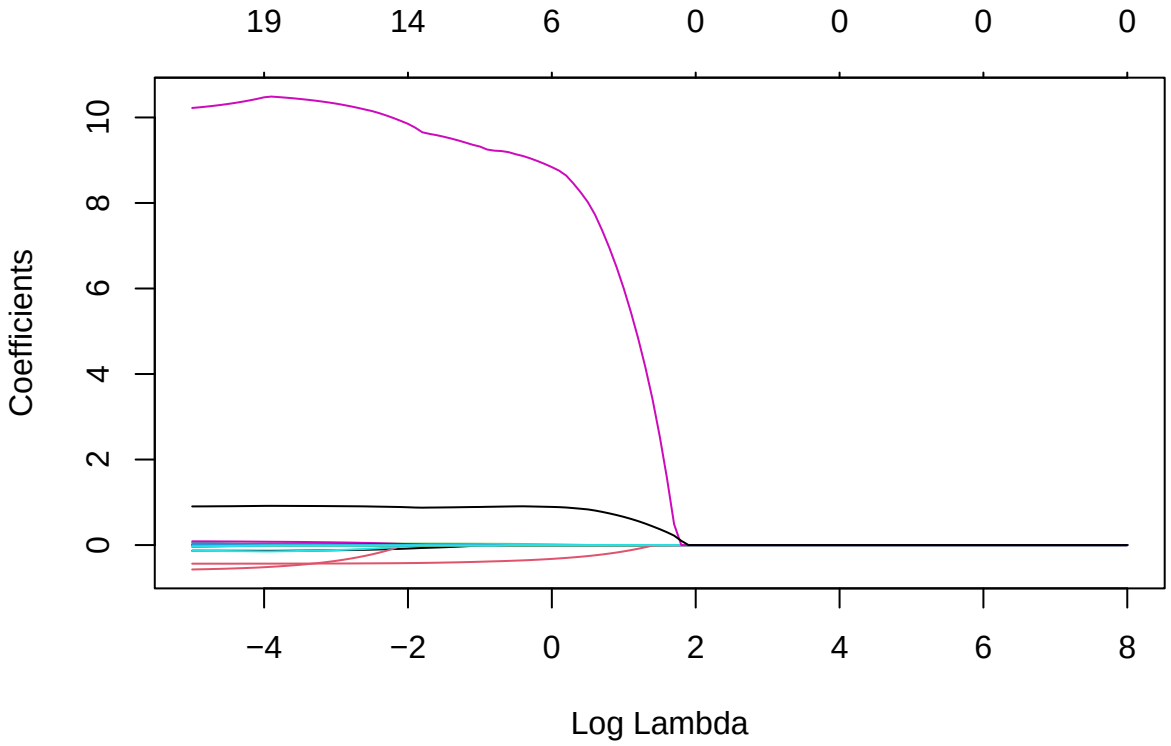
```r
#Penalty type
cv.lambda <- cv.glmnet(X, Y,
                       alpha = 1,
                       lambda=exp(seq(-5,8,.1)))

plot(cv.lambda)
```




```
cv.lambda$lambda.min
```

```
## [1] 0.006737947
```

```
# Line path
plot(cv.lambda$glmnet.fit,
     "lambda", label=FALSE)
```



minimum/best lamba is 0.006737947

```
## Final Model
lmin <- cv.lambda$lambda.min
lasso.model <- glmnet(x=X, y=Y,
                      alpha = 1,
                      lambda = lmin)
lasso.model$beta
```

```
## 20 x 1 sparse Matrix of class "dgCMatrix"
##                                s0
## year                -1.318294e-01
## statusDeveloping    -5.698184e-01
## adult_mortality     -1.734378e-02
## infant_deaths        5.607867e-02
## alcohol             -1.351466e-01
## perc_expend          2.748542e-04
## hep_b                1.812969e-03
## measles              3.758079e-06
## bmi                  3.362309e-02
## X5yr_deaths         -4.356029e-02
## polio                1.100174e-02
## tot_expend           8.680241e-02
## diphtheria           7.199883e-03
## hiv_aids            -4.358351e-01
## gdp                  3.096017e-05
## population           8.771721e-10
## thin_1to19           .
## thin_5to9           -2.898429e-02
## inc_comp_resources   1.022197e+01
## schooling            9.027644e-01
```

Part III: Comparing the two final models from above

```
loocv_finalmod
```

```
##
## Call:
## lm(formula = life_exp_yrs ~ adult_mortality + infant_deaths +
##     perc_expend + X5yr_deaths + hiv_aids + inc_comp_resources +
##     schooling + hiv_deaths_cat, data = life_df_train)
##
## Coefficients:
##                         (Intercept)                     adult_mortality
##                           53.0198208                         -0.0154130
##                        infant_deaths                        perc_expend
##                           0.0578468                          0.0005095
##                        X5yr_deaths                           hiv_aids
##                           -0.0461966                         -0.3231580
##                  inc_comp_resources                          schooling
##                           9.9338895                          0.7832876
## hiv_deaths_catUnder 1 Death/1000
```

```
##                         4.6586144
```

```
summary(loocv_finalmod)
```

```
##
## Call:
## lm(formula = life_exp_yrs ~ adult_mortality + infant_deaths +
##     perc_expend + X5yr_deaths + hiv_aids + inc_comp_resources +
##     schooling + hiv_deaths_cat, data = life_df_train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.8894  -1.9532   0.1035   1.9887  12.7761
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                        5.302e+01  6.221e-01  85.230  < 2e-16 ***
## adult_mortality                   -1.541e-02  1.075e-03 -14.341  < 2e-16 ***
## infant_deaths                      5.785e-02  1.206e-02   4.796 1.83e-06 ***
## perc_expend                        5.095e-04  6.442e-05   7.910 6.04e-15 ***
## X5yr_deaths                       -4.620e-02  9.004e-03  -5.130 3.39e-07 ***
## hiv_aids                          -3.232e-01  2.019e-02 -16.005  < 2e-16 ***
## inc_comp_resources                 9.934e+00  8.809e-01  11.276  < 2e-16 ***
## schooling                          7.833e-01  6.100e-02  12.841  < 2e-16 ***
## hiv_deaths_catUnder 1 Death/1000   4.659e+00  3.487e-01  13.360  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.404 on 1146 degrees of freedom
## Multiple R-squared:  0.8533, Adjusted R-squared:  0.8523
## F-statistic: 833.3 on 8 and 1146 DF,  p-value: < 2.2e-16
```

```
lasso.model$beta
```

```
## 20 x 1 sparse Matrix of class "dgCMatrix"
##                               s0
## year                -1.318294e-01
## statusDeveloping    -5.698184e-01
## adult_mortality     -1.734378e-02
## infant_deaths        5.607867e-02
## alcohol             -1.351466e-01
## perc_expend          2.748542e-04
## hep_b                1.812969e-03
## measles              3.758079e-06
## bmi                  3.362309e-02
## X5yr_deaths         -4.356029e-02
## polio                1.100174e-02
## tot_expend           8.680241e-02
## diphtheria           7.199883e-03
## hiv_aids            -4.358351e-01
## gdp                  3.096017e-05
## population           8.771721e-10
## thin_1to19           .
## thin_5to9           -2.898429e-02
## inc_comp_resources   1.022197e+01
```

```
## schooling           9.027644e-01
```

Comparing the final models:

- LASSO model contains all of the features
- The best subset model showed that only 7 variables were needed to minimize the MSE
- Different features had very different betas, for instance schooling's beta was 9.007 in LASSO while only being 0.74 in LOOCV.
- Best subset model did not account for multicollinearity for which some of the VIF factors were very high.
- Best subset model achieved an adjR2 value of 0.85, best among the milestones so far.

Comparing MSE's

Best subset:

```
library(caret)
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'lattice'
```

```
## The following object is masked from 'package:faraway':
##
##     melanoma
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

```
life_loocvtest<-predict(loocv_finalmod, life_df_test)
(RMSE(life_loocvtest, life_df_test$life_exp_yrs, na.rm = TRUE))^2
```

```
## [1] 12.51368
```

LASSO Regression:

Looking at the cv.lambda.lasso the minimum MSE value was 12.8 which was still higher than the best subset model. The MSE value for the minimum lambda of 0.0067 was higher around 17.

Overall, the best subset linear model was able to achieve an adjusted R2 value of 0.85 and MSE of 12.51 which was the smallest compared to the lasso model, therefore the best subset model performed most accurately.