FINDING THE BEST MODEL WITH PREDICTABILITY AND INTERPRETABILITY TO PREDICT DIABETES PROGRESSION

Ramachandran Balasubramanian, Yashita Jain, Yijun Zhou

Abstract

We utilized several supervised learning approaches and compared their performance to find the method which minimizes the error and significantly improves the overall predictability and interpretability of the model. Initially, regression models were used to quantify the relationship between 10 different predictors such as age, sex, body mass index (BMI), average blood pressure and six serum measurements and a response which measures the progression of diabetes after a year. Ordinary least squares method (OLS) identified SEX, BMI, BP and S5 to be statistically significant. Concerns about multicollinearity were indicated and as a remedy, we performed regularization methods such as Ridge Regression and Lasso. In addition, we also transformed the predictors (dimension reduction) using Partial Least Squares (PLS) and Principal Component Regression (PCR). Upon considering the prediction accuracy and interpretability, ridge regression gave the best model.

1. Exploratory Data Analysis

1.1 Summary Statistics and Correlation

To understand the spread of the data, we computed descriptive summary statistics including quantitative measures such as minimum, first quartile, median, third quartile and maximum using boxplots (Figure 1b, 1c, 1d). To identify the relationship among variables and with the response, we constructed a correlation scatterplot, which identified high correlation between variables such as S1 and S2 and among S3, S4 and S5 (Figure 1a). Also, predictors such as BMI, BP, S3-S6 seems to be highly correlated with the response (Figure 1a). The summary statistics of age (2 levels) specified the number of observations (1-235 2-207) in each level.

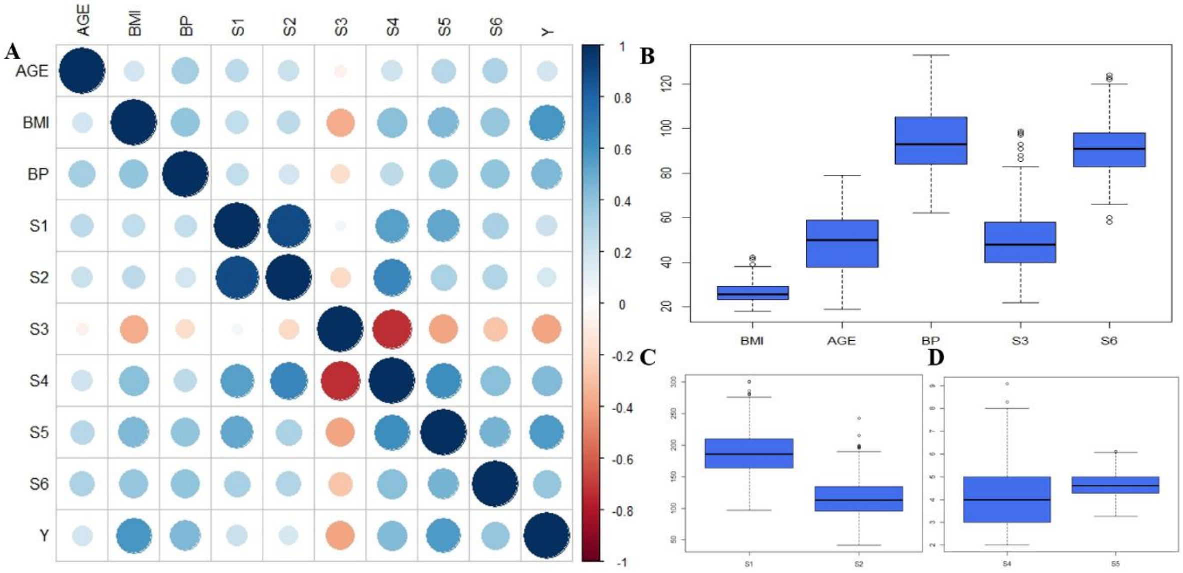
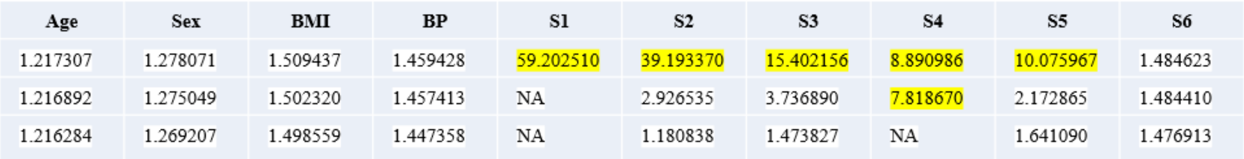


Figure 1: Summary Statistics and Correlation plot A) Correlation plot between response and predictors [Intensity of color indicates the strength of correlation. Light to dark blue - positive correlation; light to dark brown - negative correlation] **B)** Boxplot showing the summary

statistics of BMI, AGE, BP, S3, S6 **C)** S1, S2 **D)** S4, S5.

1.2 Correlation and Multicollinearity

To quantify the multicollinearity suggested by the scatterplot, we calculated variance inflation factor (VIF), which identified that variables S1 - S5 have VIF values more than 10. Hence, we removed variables S1 and S4 which resulted in a VIF less than 2 in all other predictors. We utilized the remaining predictors for subsequent model selection using Stepwise Regression.



Age	Sex	BMI	BP	S1	S2	S3	S4	S5	S6
1.217307	1.278071	1.509437	1.459428	59.202510	39.193370	15.402156	8.890986	10.075967	1.484623
1.216892	1.275049	1.502320	1.457413	NA	2.926535	3.736890	7.818670	2.172865	1.484410
1.216284	1.269207	1.498559	1.447358	NA	1.180838	1.473827	NA	1.641090	1.476913

Table 1: Variance Inflation Factor (VIF) of variables. The first row of the table indicates the VIF of all the variables (the highlighted values are considered high). After removing S1 (as indicated by NA in column 2), the VIF of other variables decrease. Finally removing S4 drops the VIF of all the predictors to less than 2.

2. Statistical Analysis and Modeling

The data was randomly split into training and test data set containing 75% and 25% of the observations respectively. The mean square error MSE (RMSE) is regarded as a criterion to detect the prediction accuracy of the model.

2.1 Ordinary Least Squares, Stepwise Regression and Best subset selection

To identify statistically significant predictors to explain the variability of the model, we started with the multiple linear regression model. The model indicated that SEX, BMI, BP and S5 (Supplemental Figure 2) are to be included in the final model and the test MSE was 3160.52. The multicollinearity of the variables and high VIF suggested that S1 and S4 must be removed from the model. To prevent the loss of information by simple elimination of these variables [1], we averaged redundant variables S1 and S2, S3 and S4 and fitted a Stepwise regression model and best subset selection (containing 8 variables). Both the model retained all the variables except age. The test MSE of the stepwise regression and best subset selection were 3217.31 and 3222.18 respectively.

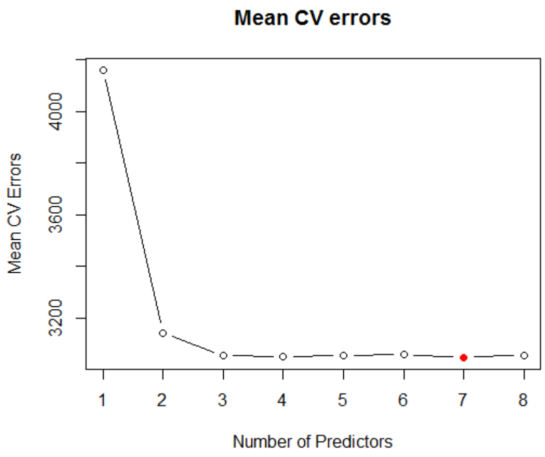


Figure 2: Best Subset selection. The above graph indicates that the 10-fold cross validation error is lowest when the number of predictors equals 7.

2.2 Ridge regression and Lasso

To address the multicollinearity mentioned earlier, we utilized regularization methods such as Ridge regression and Lasso. Tuning parameter (λ) was chosen by 10-fold cross validation in ridge regression and lasso. The test MSE of the ridge regression model was 3078. 6 which was lesser than that of lasso model which was 3138.92. Lasso included four predictors (BMI, BP, S3 and S5) in the final model and ridge regression included all the variables as expected.

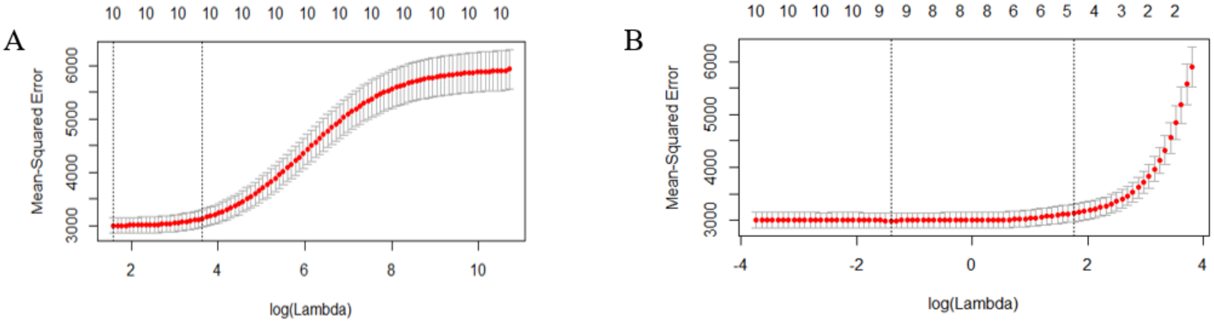
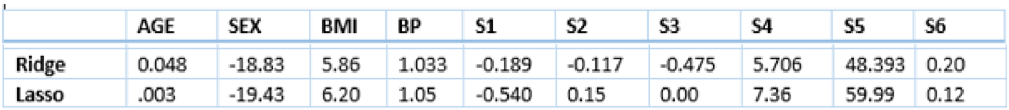


Figure 3: Ridge Regression and Lasso cross-validation curves. A) The graph depicts the cross-validation curve obtained using ridge regression (red-dotted line) along with upper and lower standard deviation curves along the lambda sequence (error bars) **B)** The graph depicts the cross-validation curve obtained using Lasso (red-dotted line). The vertical dotted lines indicate the two selected lambdas.



	AGE	SEX	BMI	BP	S1	S2	S3	S4	S5	S6
Ridge	0.048	-18.83	5.86	1.033	-0.189	-0.117	-0.475	5.706	48.393	0.20
Lasso	.003	-19.43	6.20	1.05	-0.540	0.15	0.00	7.36	59.99	0.12

Table 2: Coefficients of predictors. The above table summarizes the coefficients of each predictor obtained using Ridge regression and Lasso.

2.3 Principal Component Regression (PCR) and Partial least squares (PLS)

Principal component regression was carried out and then 10-fold cross-validation error was calculated was each possible principal component (M) used. The smallest cross-validation error occurred when seven (M=7) components were used (Figure 4a). The test MSE is 3128.946 which is higher than ridge regression but slightly lower than lasso. Nevertheless, cross validation error was lower from M=4 (83% of the variability in the model explained, Supplemental Figure 3), suggesting that smaller number of components would suffice to explain the model.

Similarly, in partial least squares, the lowest cross validation error occurred when M=3 components were used (Figure 4b) and the corresponding test MSE is 3090.89.

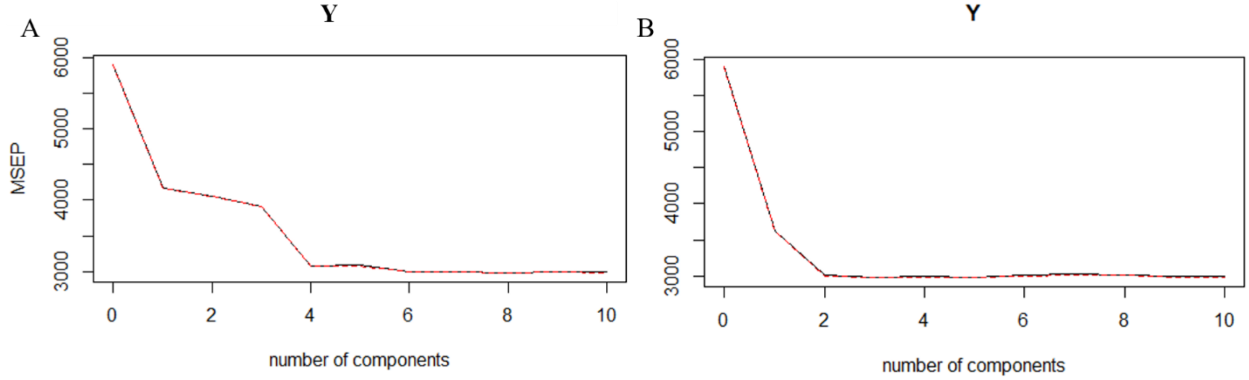


Figure 4: PCR and PLS. Graph depicting 10-fold CV error for each principal component A) Principal component regression (PCR) shows that M=7 components are required to attain the lowest test MSE B) Partial least squares (PLS) shows that M=4 components are required to explain 63.73% of the variability (Supplemental figure 4). The black solid line indicates cross-validation error and red dashed line indicates the adjusted cross validation error.

3. Discussion

To understand the general spread of the data, we constructed boxplots which helped us understand the descriptive statistics of the dataset (Figure 1b, 1c and 1d, Supplemental Figure 1). The scatterplot correlation plot identified several key correlations among variables and predictors (Figure 1a). Higher VIF values suggested that the variables S1 and S4 need to be removed from the final model (Table 1). However, averaging the redundant variables and fitting the model may preserve the loss of valuable information from the data [1]. Hence we averaged the redundant variables S1 and S2, S3 and S4, so that the information is not lost from eliminating either of them. The resulting model was fit using Stepwise regression which included all the variables in the model except age, which was consistent with the result obtained from best subset selection (Figure 2). In addition, we performed best subset selection and stepwise regression after removing variables S1 and S4, and we found that the best subset selection resulted in the lowest test MSE of 2661.05 with variables SEX, BMI, BP, S2, S3 and S5 (Supplemental figure 5). Stepwise selection chose all the five variables as best subset selection. We initially were convinced that this is the best model, until we found simply omitting variables must result in a model which is not cogent.

However, regularization methods such as ridge regression and lasso, can be utilized for dealing with multicollinearity. Ridge regression resulted in a test MSE of 3078.6, which outperformed all the models considered in this study (except the bets subset and stepwise regression models without S1 and S4). Hence we regard this as the best model in terms of prediction accuracy.

Additionally, utilizing dimension reduction methods, resulted in a three-component model using PLS and seven component model using PCR. The significant reduction in dimension in PLS can be attributed to the ability of PLS to search for directions that explain the variance in both the response and the predictors, while PCR attempts to maximize the amount of variance explained by the dependent variables. The performance of both PLS and PCR models were comparable, but the lack of specific predictors makes the interpretation of the model very difficult.

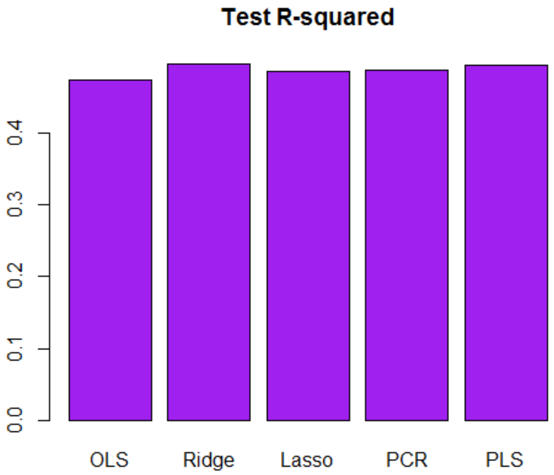


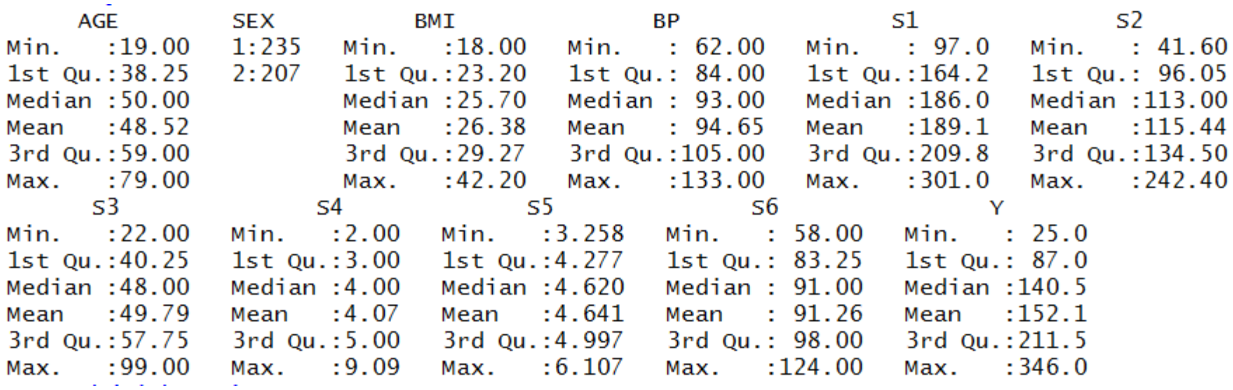
Figure 5: Comparison of adjusted R-square from various models. Upon comparing the R-square values, the ridge regression seems to explain the maximum variability of the model, which in accordance with its lowest test MSE, followed by PLS. PCR and Lasso have similar values and OLS has the least value among all the methods. This corroborates our conclusion that Ridge regression is the best model to explain this data.

REFERENCES

[1] web.pdx.edu/~newsomj/da2/ho_remedies.pdf

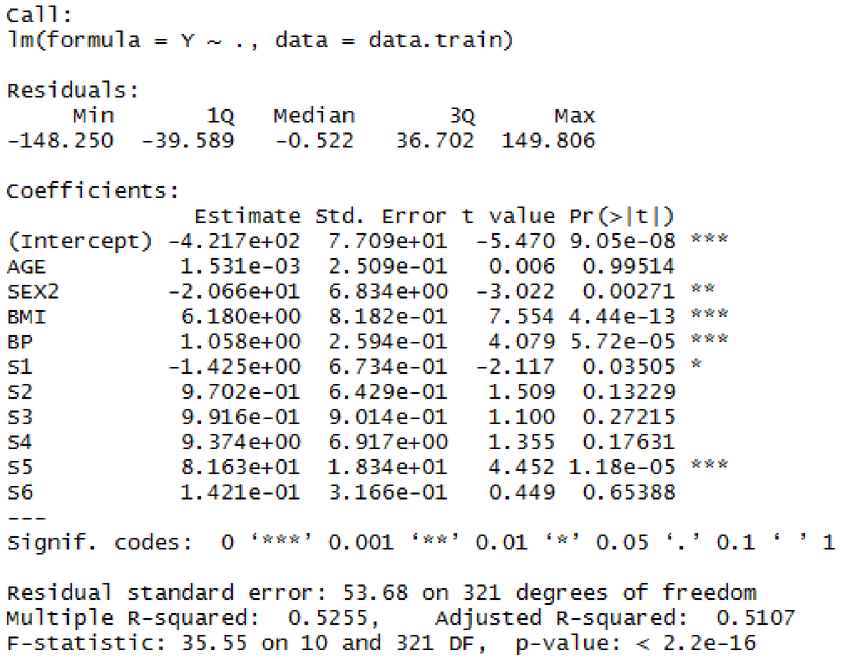
APPENDIX

1. Supplement Tables and Figures



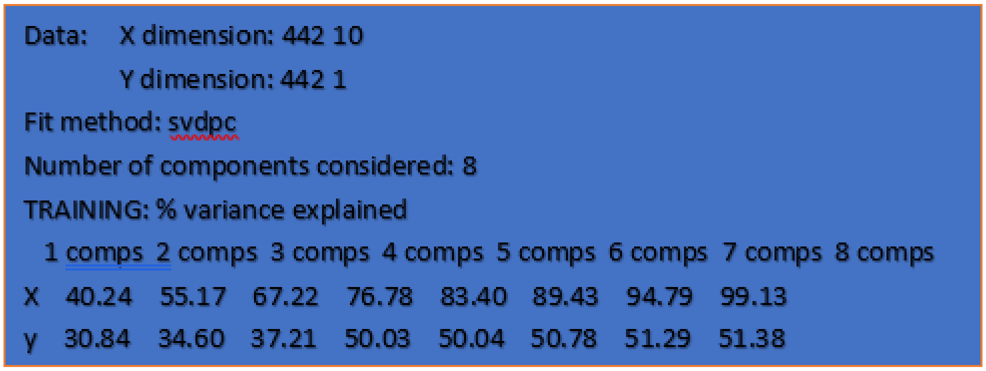
```
      AGE          SEX          BMI               BP                 S1               S2
 Min.   :19.00   1:235    Min.   :18.00   Min.   : 62.00   Min.   : 97.0   Min.   : 41.60
 1st Qu.:38.25   2:207    1st Qu.:23.20   1st Qu.: 84.00   1st Qu.:164.2   1st Qu.: 96.05
 Median :50.00            Median :25.70   Median : 93.00   Median :186.0   Median :113.00
 Mean   :48.52            Mean   :26.38   Mean   : 94.65   Mean   :189.1   Mean   :115.44
 3rd Qu.:59.00            3rd Qu.:29.27   3rd Qu.:105.00   3rd Qu.:209.8   3rd Qu.:134.50
 Max.   :79.00            Max.   :42.20   Max.   :133.00   Max.   :301.0   Max.   :242.40
       S3              S4              S5               S6                Y
 Min.   :22.00   Min.   :2.00   Min.   :3.258   Min.   : 58.00   Min.   : 25.0
 1st Qu.:40.25   1st Qu.:3.00   1st Qu.:4.277   1st Qu.: 83.25   1st Qu.: 87.0
 Median :48.00   Median :4.00   Median :4.620   Median : 91.00   Median :140.5
 Mean   :49.79   Mean   :4.07   Mean   :4.641   Mean   : 91.26   Mean   :152.1
 3rd Qu.:57.75   3rd Qu.:5.00   3rd Qu.:4.997   3rd Qu.: 98.00   3rd Qu.:211.5
 Max.   :99.00   Max.   :9.09   Max.   :6.107   Max.   :124.00   Max.   :346.0
```

Figure 1: Summary statistics of the diabetes data. The above figure provides quantitative measures of minimum, first quartile, median, mean, third quartile and maximum.



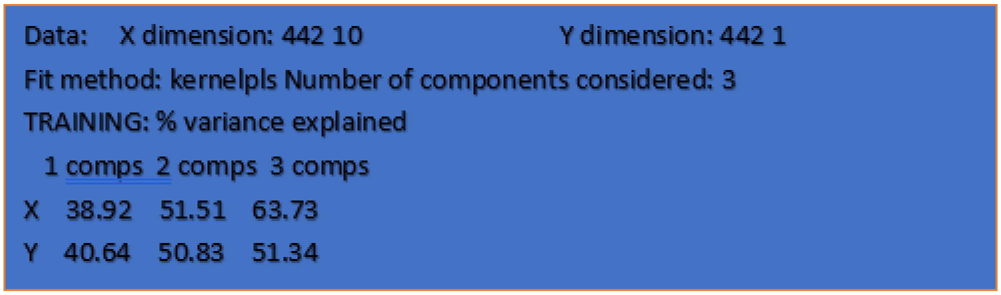
```
Call:
lm(formula = Y ~ ., data = data.train)

Residuals:
     Min       1Q   Median       3Q      Max 
-148.250  -39.589   -0.522   36.702  149.806 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) -4.217e+02  7.709e+01  -5.470 9.05e-08 ***
AGE          1.531e-03  2.509e-01   0.006  0.99514    
SEX2        -2.066e+01  6.834e+00  -3.022  0.00271 ** 
BMI          6.180e+00  8.182e-01   7.554 4.44e-13 ***
BP           1.058e+00  2.594e-01   4.079 5.72e-05 ***
S1          -1.425e+00  6.734e-01  -2.117  0.03505 *  
S2           9.702e-01  6.429e-01   1.509  0.13229    
S3           9.916e-01  9.014e-01   1.100  0.27215    
S4           9.374e+00  6.917e+00   1.355  0.17631    
S5           8.163e+01  1.834e+01   4.452 1.18e-05 ***
S6           1.421e-01  3.166e-01   0.449  0.65388    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 53.68 on 321 degrees of freedom
Multiple R-squared:  0.5255,	Adjusted R-squared:  0.5107 
F-statistic: 35.55 on 10 and 321 DF,  p-value: < 2.2e-16
```

Figure 2: Intercept values obtained using Ordinary Least squares (OLS). SEX2, BMI, BP and S5 are significant.



```
Data:    X dimension: 442 10
         Y dimension: 442 1
Fit method: svdpc
Number of components considered: 8
TRAINING: % variance explained
  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps  8 comps
X   40.24    55.17    67.22    76.78    83.40    89.43    94.79    99.13
y   30.84    34.60    37.21    50.03    50.04    50.78    51.29    51.38
```

Figure 3: Variability explained by PCR model.



```
Data:    X dimension: 442 10                    Y dimension: 442 1
Fit method: kernelpls Number of components considered: 3
TRAINING: % variance explained
  1 comps  2 comps  3 comps
X   38.92    51.51    63.73
Y   40.64    50.83    51.34
```

Figure 4: Variability explained by PLS model.

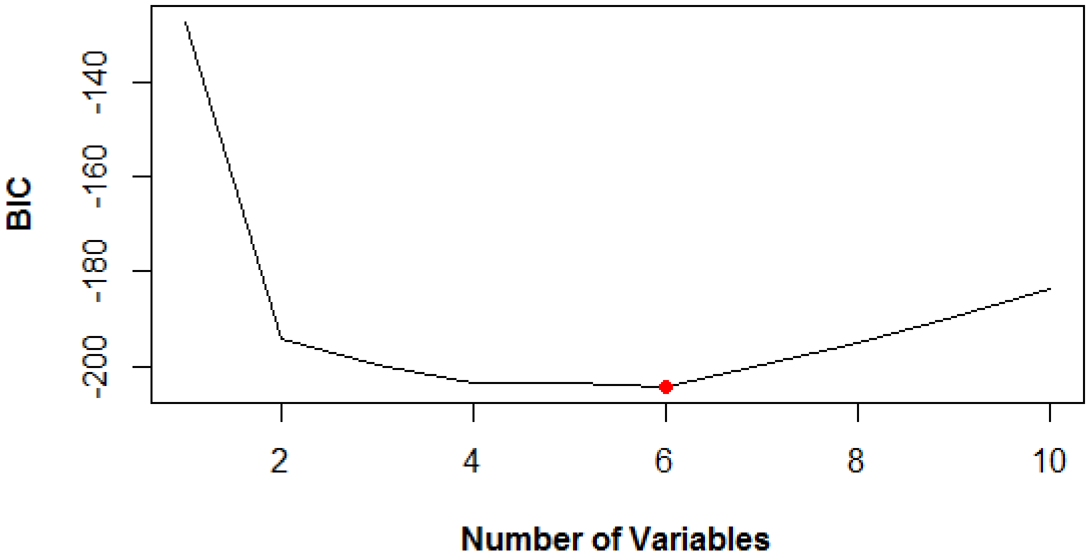


Figure 5: Best subset selection using BIC selected 6 predictor model including SEX, BMI, BP, S2, S3 and S5. (after removing S1 and S4 which had high VIF)

2. R Code
1. Exploratory Data Analysis
1.1 Summary Statistics and Correlation

##Reading the data

```
Diabetes = read.table("diabetes.data.txt", header=TRUE)
Diabetes$SEX = as.factor(Diabetes$SEX)
summary(Diabetes)
attach(Diabetes)
anyNA(Diabetes)
SEX = as.factor(SEX)
```

##Exploratory Data Analysis

```
install.packages("corrplot")
library(corrplot)
cormax=cor(Diabetes[-2])
corrplot(cormax, method="circle",tl.col="black")
boxplot(BMI,AGE, BP, S3, S6, col="royalblue2", names=c("BMI", "AGE", "BP", "S3", "S6"))
boxplot(S1,S2, col="royalblue2", names=c("S1", "S2"))
boxplot(S4,S5, col="royalblue2", names=c("S4", "S5"))
plot(SEX, col=c("royalblue2", "violet"), xlab="SEX",font.axis=4)

diabetes=read.table("diabetes.data.txt",header=TRUE)
```

2. Data Modeling

```
summary(diabetes)
diabetes$SEX=as.factor(diabetes$SEX)
set.seed(26)
n=nrow(diabetes)

test <- sample(n, round(n/4))
data.train <- diabetes[-test,]
data.test <- diabetes[test,]
x <- model.matrix(Y~ ., data = diabetes)[,-1]

x.train <- x[-test,]
x.test <- x[test,]
y <- diabetes$Y
y.train <- y[-test]
y.test <- y[test]
```

#OLS

```
ols.lm=lm(Y~.,data.train)
round(coef(summary(lm(Y~., data = data.train))),2)
coefi  = coef(ols.lm,id=10)
lm.pred   = coefi[1] + (x.test[, names(coefi[-1])]%*%coefi[-1])
cat(paste0("The test MSE for the OLS model is: ",round(mean((y.test - lm.pred)^2),2)))
summary(ols.lm)
```

#stepwise regression

```
S12=(scale(diabetes$S1)+scale(diabetes$S2))/2
S34=(scale(diabetes$S3)+scale(diabetes$S4))/2
drops=c("S1","S2","S3","S4")
DS=diabetes[ , !(names(diabetes) %in% drops)]
DS=data.frame(DS,S12,S34)
DS.train <- DS[-test,]
DS.test <- DS[test,]
x <- model.matrix(Y~ ., data = DS)[,-1]

x.train <- x[-test,]
x.test <- x[test,]
y <- DS$Y
y.train <- y[-test]
regfit.full = lm(Y~., data = DS.train)
step <- stepAIC(regfit.full, direction="both")
regfit.full = lm(Y~.-(S12+S6+AGE), data = DS.train)
round(coef(summary(lm(Y~.-(S12+S6+AGE), data = DS.train))),3)
coefi  = coef(regfit.full,id=5)
sr.pred   = coefi[1] + (x.test[, names(coefi[-1])]%*%coefi[-1])
cat(paste0("The test MSE for Step-wise regression model is: ",round(mean((y.test - sr.pred)^2),2)))

#best_subset with 10 fold
predict.regsubsets = function(object, newdata, id,...){
 form=as.formula(object$call[[2]])
 mat = model.matrix(form, newdata)
 coefi = coef(object, id=id)
 xvars = names(coefi)
 mat[,xvars]%*%coefi
}

k=10
set.seed(26)
folds = sample(1:k, nrow(DS.train),replace = TRUE)
cv.errors = matrix(NA,k,8, dimnames = list(NULL, paste(1:8)))

for (j in 1:k){
 best.fit = regsubsets(Y~. ,data=DS.train[folds!=j,],nvmax = 8)
 for (i in 1:8){
        k.pred = predict.regsubsets(best.fit, DS.train[folds==j,], id=i)
        cv.errors[j,i] = mean((DS.train$Y[folds==j]-k.pred)^2)
 }
}

mean.cv.errors = apply(cv.errors, 2, mean)
plot(mean.cv.errors, type="b", main = "Mean CV errors",xlab = "Number of Predictors",
        ylab="Mean CV Errors")


y = min(mean.cv.errors)
x = which.min(mean.cv.errors)
```

```
points(x,y, col="red", cex=1, pch=19)

regfit.cv = regsubsets(Y~. , data = DS.train , nvmax = 8)
coefi  = coef(regfit.cv,id=7)
cv.pred   = coefi[1] + (x.test[, names(coefi[-1])]%*%coefi[-1])
coefi
cat(paste0("The test MSE for the best subsets model using CV is: ",round(mean((y.test -
cv.pred)^2),2)))

#ridge regression
library(glmnet)
grid = 10^seq(10,-2,length=100)
set.seed(26)
cv.out = cv.glmnet(x.train, y.train, alpha = 0)
plot(cv.out)
largelam = cv.out$lambda.min
ridge.mod = glmnet(x.train, y.train, alpha = 0, lambda = grid, thresh = 1e-12)
ridge.pred = predict(ridge.mod, s=largelam, newx=x.test)
cat(paste0("The test MSE for the ridge regression model using CV is: ",round(mean((y.test -
ridge.pred)^2),2)))
coef = glmnet(x.train, y.train, alpha = 0, lambda = largelam, thresh = 1e-12)$beta
matrix(coef, dimnames = list(row.names(coef), c("Coefficient")))

#lasso
set.seed(26)
cv.out = cv.glmnet(x.train, y.train, alpha = 1)
plot(cv.out)
largelam = cv.out$lambda.min
lasso.mod = glmnet(x.train, y.train, alpha = 1, lambda = largelam)
lasso.pred = predict(lasso.mod, s=largelam, newx = x.test)
cat(paste0("The test MSE for the lasso model using CV is: ",round(mean((y.test -
lasso.pred)^2),2)))

coef= glmnet(x.train, y.train, alpha = 1, lambda = largelam)$beta
matrix(coef, dimnames = list(row.names(coef), c("Coefficient")))

#pcr
library(pls)
set.seed (26)
pcr.fit=pcr(data.train$Y~., data=data.train,scale=TRUE ,validation ="CV")
validationplot(pcr.fit ,val.type="MSEP")
summary(pcr.fit)
pcr.pred=predict(pcr.fit ,x.test, ncomp =8)
mean((pcr.pred -y.test)^2)
```

```
pcr.fit=pcr(y~x,scale =TRUE ,ncomp=8)
summary(pcr.fit)


#pls
set.seed(26)
pls.fit=plsr(Y~., data=data.train,scale=TRUE ,validation ="CV")
summary (pls.fit )
validationplot(pls.fit ,val.type="MSEP")
pls.pred=predict (pls.fit ,x.test, ncomp =3)
mean((pls.pred -y.test)^2)
pls.fit=plsr(Y~., data=diabetes,scale=TRUE ,ncomp =3)
summary(pls.fit)

#comparison
test.avg = mean(data.test[, "Y"])
lm.test = 1 - mean((data.test[, "Y"] - lm.pred)^2) /mean((data.test[, "Y"] - test.avg)^2)
ridge.test = 1 - mean((data.test[, "Y"] - ridge.pred)^2) /mean((data.test[, "Y"] - test.avg)^2)
lasso.test = 1 - mean((data.test[, "Y"]- lasso.pred)^2) /mean((data.test[, "Y"] - test.avg)^2)
pcr.test = 1 - mean((data.test[, "Y"] - data.frame(pcr.pred))^2) /mean((data.test[, "Y"] - test.avg)^2)
pls.test = 1 - mean((data.test[, "Y"] - data.frame(pls.pred))^2) /mean((data.test[, "Y"] - test.avg)^2)
barplot(c(lm.test, ridge.test, lasso.test, pcr.test, pls.test), col="purple", names.arg=c("OLS", "Ridge", "Lasso", "PCR", "PLS"), main="Test R-squared")
```