MAS3907 Group Project

Lingyu Tan, Man Kuan Io

Contents

1 Introduction

In this project, we are trying to find a linear regression model that best fits the *Boston* data set given in the *nclSLR* package for the purpose of discovering which variables influenced the crime rate in Boston. We will only use 400 items that are randomly generated from the data set using seed *18053836*.

2 Exploratory Data Analysis

The first thing we do is to take a look at the class of each column of the data. As there are only 14 columns and all the columns are variables, we can generate a scatterplot matrix using all the variables as below:

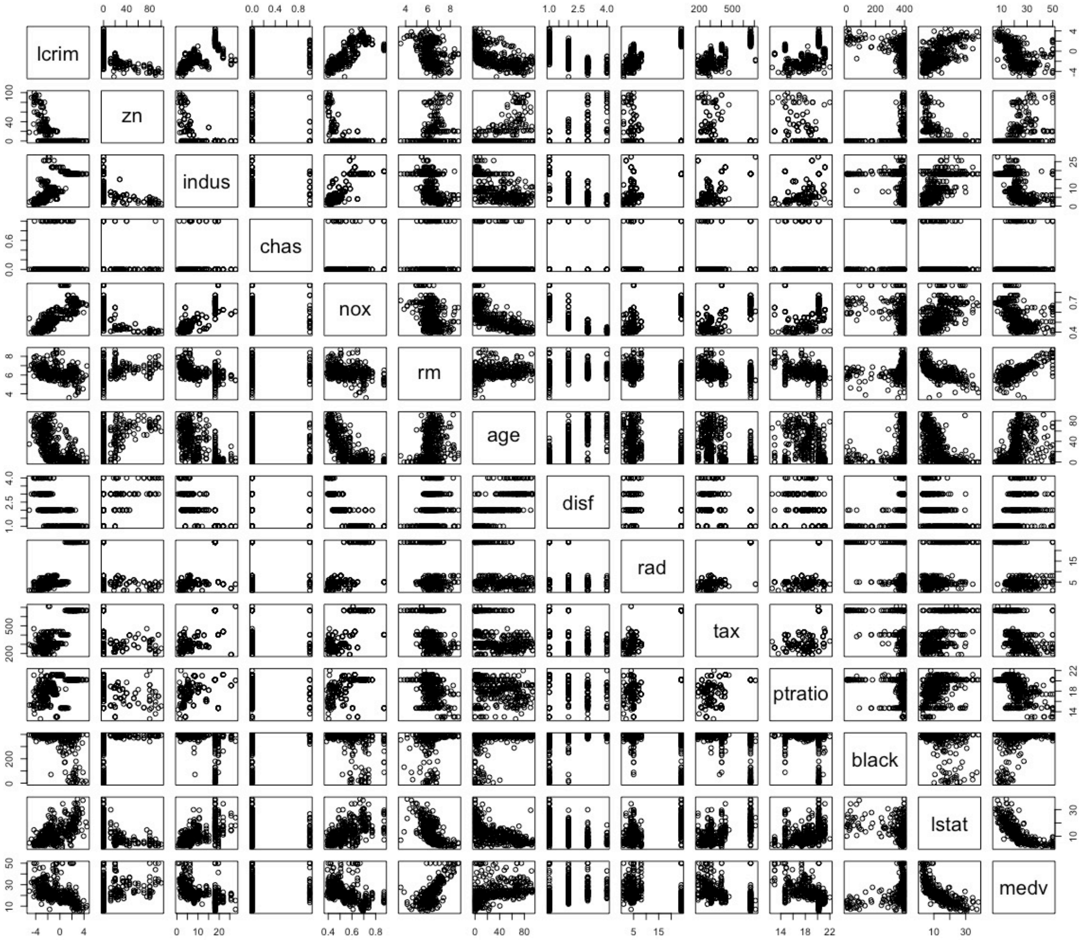

Figure 1: Scatterplot matrix for Boston data

By observing the scatter graphs of *lcrim* against the other 13 variables, we see that:

- There appears to be a strong positive linear relationship between *lcrim* and *nox*, *lstat* with correlation coefficients being 0.790 and 0.648 respectively (See Appx. A).

- There seems to be no clear patterns in the scatterplots of *lcrim* against *rad* and *tax*; however, a cluster of high-leverage points at the top-right corner in both plots may lead to a highly positive correlation (0.855, 0.848 respectively).
- *lcrim* is negatively correlated with *zn, black, medv, age, disf* (-0.529, -0.472, -0.484, -0.668, -0.705 respectively).

By observing the scatterplots of 13 predictor variables against each other, we have:

- There appears to be a negative linear relationship between *rm* and *lstat* (-0.613) and a clear positive relationship between *rm* and *medv* (0.703), thus a multicollinearity is suggested and we probably do not need to include all these three variables in our model.
- *lstat* is highly negatively correlated with *medv* (-0.738).

Taking a closer look at the correlation matrix, we see that:

- The correlation of *tax* and *rad* is 0.925 indicating a highly significant positive correlation, so we might only need one of them in our model.
- The correlation coefficients between *chas* and other variables are all fairly close to zero, being only -0.006 between *chas* and *lcrim*, which means we probably do not need *chas* in our model.

3 Treatment of Variable *disf*

We know that the variable *disf* is an ordered categorical variable with four levels indicating the weighted mean of the distances to five Boston employment centres. The larger the weighted mean is, the larger the *disf* will be. Considering that *disf* is generated from a continuous variable, it could probably be treated as a quantitative variable. To verify this, we could look at the scatterplot of *lcrim* against *disf* below:

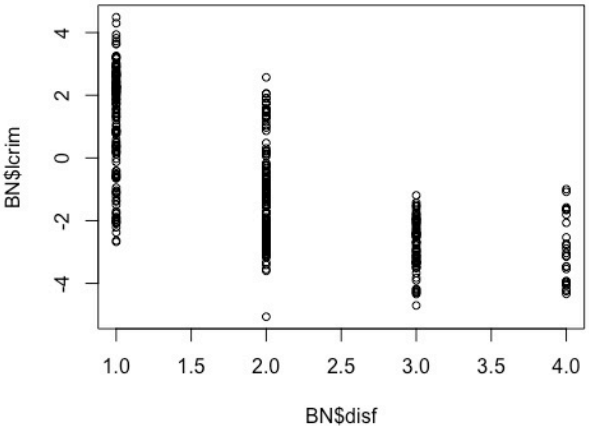

Figure 2: Scatterplot of *lcrim* against *disf*

From Figure 2, there appears to be a strong negative linear relationship between *lcrim* and *disf*. We could try to fit a linear model to see what will happen, here are the coefficients of the fitted model:

Coefficients	Estimate	Std. Error	t-value	p-value
(Intercept)	2.60099	0.18513	14.05	$<$ 2e-16 ***
BN$disf	-1.70318	0.08583	-19.84	$<$ 2e-16 ***

Table 1: Coefficients for the fitted model of *lcrim* against *disf*

We see that the estimated coefficient for *disf* is negative and the corresponding p-value is considerably small, thus a very strong negative linear relationship could be suggested.

Overall, *disf* is ordered and any effect on *lcrim* is very likely to be monotonic (more specifically, negatively correlated). We could simply represent it as a quantitative variable for convenience.

4 Model Building

In this section we use the standardised data of the 400 items mentioned above, renamed as *Boston_data*, to fit three best models by best subset selection, ridge regression and partial least squares respectively. Then we compare the MSEs using 10-fold cross validation to choose a final best model amongst these three models. In order to make the comparison fair, we define *fold_index* and *fold_list* (see Appx. B line 55-69) using the same partition, which are used in the arguments of cross validation functions.

4.1 Best Subset Selection

As we only have $p = 13$ explanatory variables leading to $2^p = 2^{13} = 8192$ possible models which is fairly small and acceptable, a best subset selection could be used. Applying that in R we will get 13 best models containing distinct number of variables identified as $M_1, M_2, ..., M_{13}$ (M_0 with no variables omitted). To choose the best one amongst these models, we can compare their adjusted R^2, Mallow's C_p and the BIC using Figure 3 below:

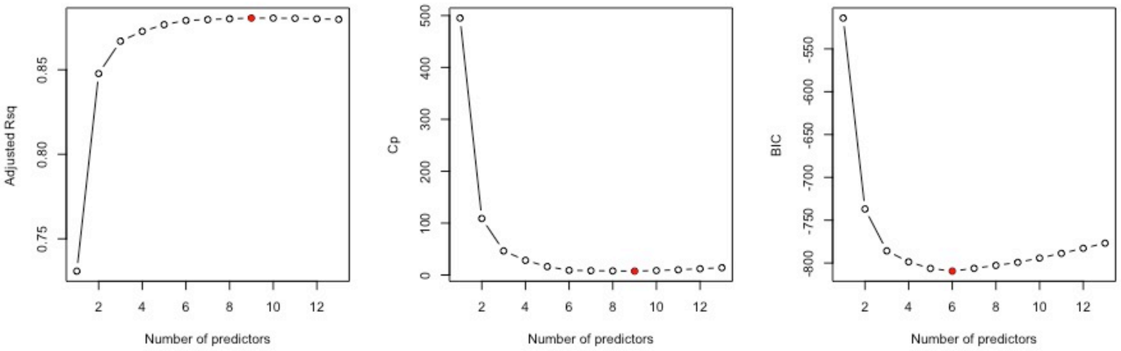

Figure 3: Best subset selection for *Boston_data*

As we can see, both adjusted R^2 and Mallow's C_p statistic suggest M_9, whilst the BIC suggests M_6. However, in the plots for adjusted R^2 and Mallow's C_p statistic, there seems to be little difference between models M_6,...,M_{13}. Therefore we might regard the best model as M_6 (which includes *zn, nox, disf, rad, black* and *lstat*).

Upon doing a 10-fold cross validation to confirm our choice, we have Table 2 of weighted average MSE for each model $M_1, M_2, ..., M_{13}$ and Figure 4 to compare:

Model	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8	M_9
MSE	1.365	0.730	0.642	0.622	0.614	0.589	0.605	0.603	0.602

Model	M_{10}	M_{11}	M_{12}	M_{13}
MSE	0.603	0.603	0.603	0.603

Table 2: Table of weighted average MSE for $M_1, M_2, ..., M_{13}$

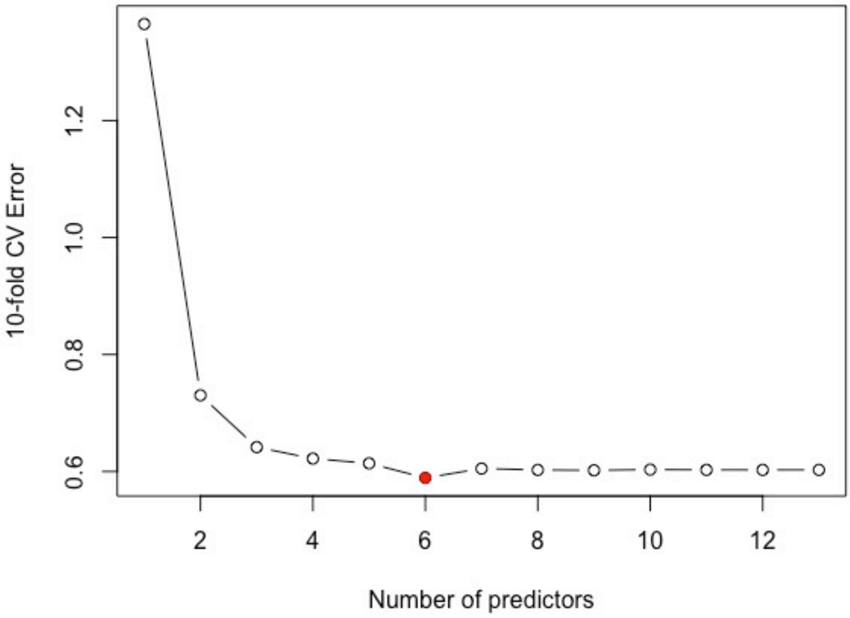

Figure 4: Comparison of cross validation errors

Obviously, M_6 has the lowest MSE indicating it is the best model by cross validation which approves our choice above. Therefore, by best subset selection we choose M_6 as the best model. The coefficients for M_6 is as follows:

Variable	(Intercept)	zn	nox	disf	rad	black	lstat
Coefficients	-0.733	-0.265	0.510	-0.238	1.206	-0.130	0.194

Table 3: Table of coefficients for M_6

From Table 3, we see that variables *indus, chas, rm, age, tax, ptratio, medv* have been removed from the best model. The coefficients of *zn, disf, black* are negative whilst those of *nox, rad, lstat* are positive.

When the proportion of residential land zoned for lots over 25,000 sq.ft. or weighted mean of the distances to five Boston employment centres goes up, the crime rate will decrease. When nitrogen oxides concentration, index of accessibility to radial highways or lower status of the population (percent) goes up, the crime rate is likely to increase.

We notice that the variable *black* is defined as $1000(Bk - 0.63)^2$, where Bk is the proportion of blacks by town. Without loss of generality (the proportion of blacks in a town is not likely to be above 0.63), we could assume Bk is less than 0.63. Considering the function is monotonically decreasing within $(0, 0.63)$, we therefore conclude that as the proportion of blacks by town goes up, the crime rate will also increase.

4.2 Ridge Regression

To determine the value of $\widetilde{\lambda}$ in the model, we use cross-validation to calculate the corresponding MSE values and the scores are plotted in Figure 5 below:

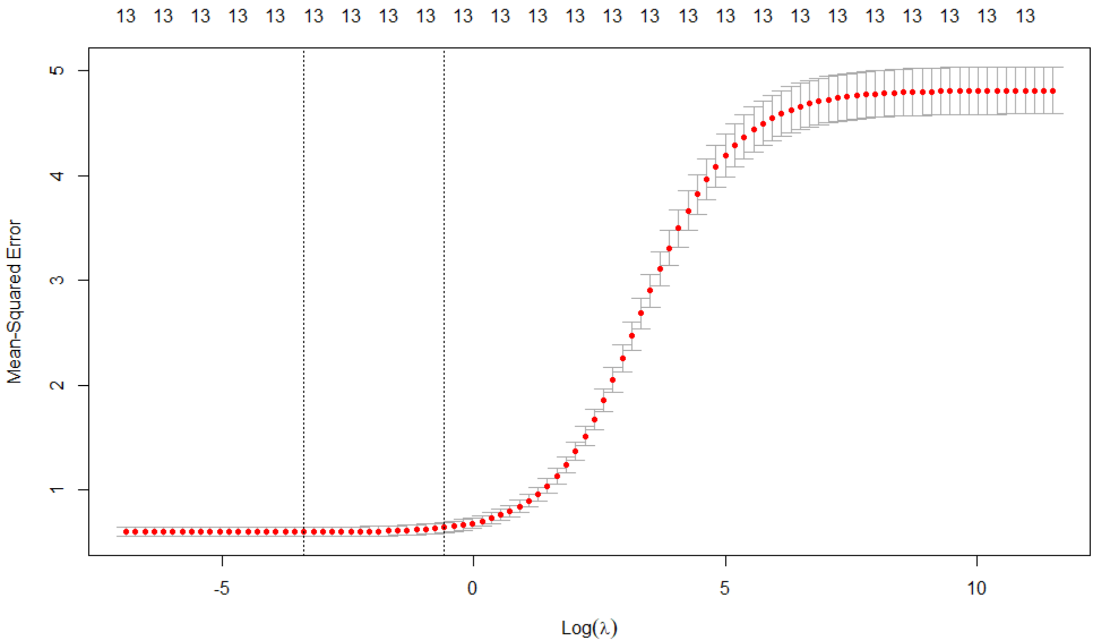

Figure 5: Cross-validation scores for *Boston_data* using Ridge Regression

From the graph, we can see that MSE minimises at the point of the first dotted line, where $\widetilde{\lambda} = 0.034$. We noticed that the chosen value of $\widetilde{\lambda}$ is very close to zero, which means that the best solution we obtained using cross-validation is very close to the results of a least squares regression.

We can see the effect on $\widetilde{\lambda}$ on the variables from Figure 6 shown below:

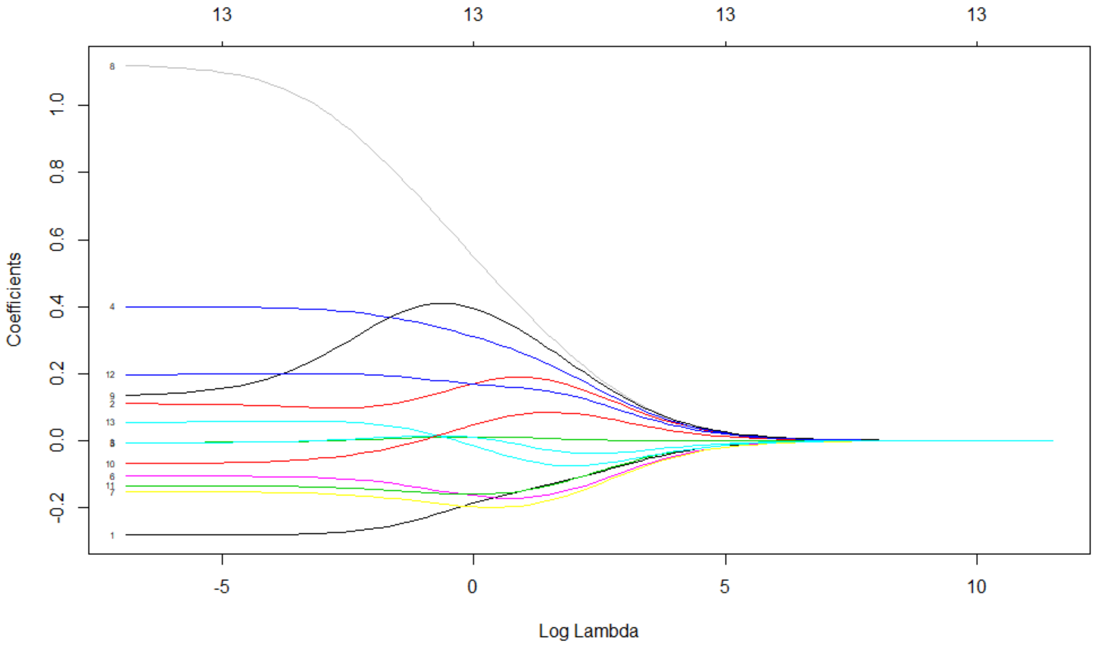

Figure 6: Graphical illustration of the effect of varying the tuning parameter

Compared with the model by best subset selection, we can see that the variables in the ridge regression model have similar coefficients to the best subset selection model and the rest of the coefficients are close to zero.

Variable	(Intercept)	zn	indus	chas	nox	rm	age
Coefficients	-0.733	-0.279	0.100	-0.002	0.393	-0.002	-0.109

Variable	disf	rad	tax	ptratio	black	lstat	medv
Coefficients	-0.157	1.022	0.225	-0.057	-0.138	0.200	0.057

Table 4: Ridge Regression for *Boston_data*

4.3 Partial Least Squares

Using PLS we could calculate the transformed predictor variables (See Appx. C). We see that:

- The first transformed variable is a weighted average of the original standardised explanatory variables except for *chas* which contributes little to the component.
- The second transformed variable is influenced mostly by *rm* and *medv*, whilst assigns little weight to *indus*, *age* and *disf*.

Moreover, the percentage of variance explained in each model can be calculated by summarising the model fit which is listed below:

Model	1 comps	2 comps	3 comps	4 comps	5 comps	6 comps	7 comps
X	48.51	58.52	65.76	72.39	77.55	80.77	84.39
y	80.69	86.68	87.67	88.09	88.18	88.29	88.35

Model	8 comps	9 comps	10 comps	11 comps	12 comps	13 comps
X	87.08	90.21	95.34	96.99	98.45	100.00
y	88.37	88.37	88.37	88.37	88.37	88.37

Table 5: Table of weighted average MSE for $M_1, M_2, ..., M_{13}$

To determine the number m of transformed explanatory variables in the model, we use cross validation to calculate corresponding MSE values and the scores are plotted in Figure 7 below:

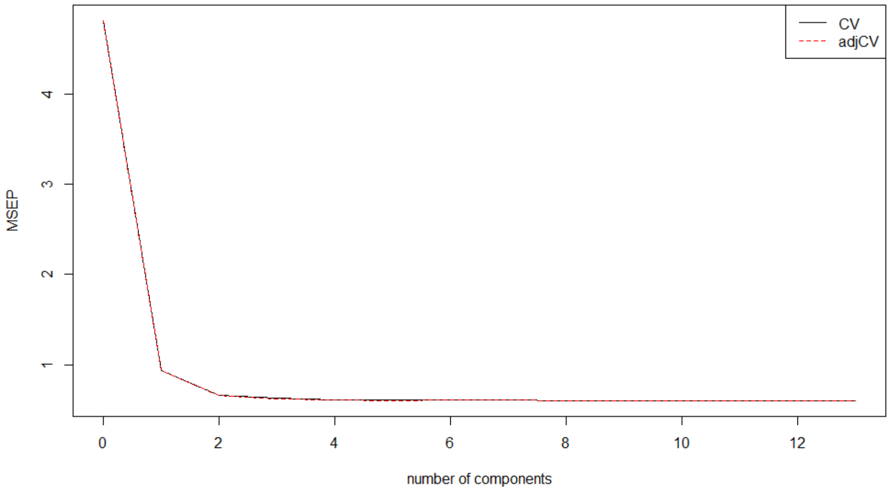

Figure 7: Cross-validation scores for *Boston_data* using PLS

In this case there is a clear elbow at two components, so we could choose $m = 2$ giving a model that uses the first two PLS directions for our transformed explanatory variable. On the scale of the original variables, the coefficients are given below:

Variable	(Intercept)	zn	indus	chas	nox	rm	age
Coefficients	-0.733	-0.141	0.227	0.036	0.347	0.115	-0.203

Variable	disf	rad	tax	ptratio	black	lstat	medv
Coefficients	-0.247	0.589	0.484	0.059	-0.218	0.128	0.037

Table 6: Coefficients for *Boston_data* using PLS

Compared with the model by best subset selection, we see that all the common variables used in both models share the same direction, which means the conclusions drawn by best subset selection also work in this model. Besides, *chas*, *rm*, *ptratio* and *medv* have relatively small coefficients in this model indicating less impact. In addition, from Table 5 we know that we are able to explain 86.68% of the variation in $\underline{y}$ using just two PLS directions.

4.4 Model Determination by Cross-validation

We can then formally compare the three chosen models using k-fold cross-validation. As mentioned above, we have chosen $k = 10$ and applied the same folds to all the models to ensure that the comparison is fair. In this context, the best model will be the model with the lowest mean-squared error.

Model	BSS	Ridge	PLS
MSE	0.589	0.685	0.603

Table 7: Cross-validation for *Boston_data*

From Table 7, we can see that out of the three models, best subset selection has the lowest mean-squared error.

5 Discussion

Throughout the process of model building, we use only 400 items out of 506 items in total. We could use the additional 106 items to perform out-of-sample validation, that is, to compute the test error and thus evaluating the performance of the above three models selected by best subset selection, ridge regression, and PLS respectively.

A Correlation Matrix

Corr.	lcrim	zn	indus	chas	nox	rm	age	disf	rad	tax	ptration	black	lstat	medv
lcrim	1.000	-0.529	0.746	-0.006	0.790	-0.320	-0.668	-0.705	0.855	0.848	0.427	-0.472	0.648	-0.484
zn	-0.529	1.000	-0.534	-0.031	-0.516	0.319	0.573	0.617	-0.316	-0.323	-0.382	0.176	-0.412	0.366
indus	0.746	-0.534	1.000	0.027	0.771	-0.417	-0.650	-0.737	0.599	0.710	0.391	-0.353	0.628	-0.516
chas	-0.006	-0.031	0.027	1.000	0.045	0.062	-0.074	-0.069	-0.039	-0.074	-0.125	0.080	-0.048	0.194
nox	0.790	-0.516	0.771	0.045	1.000	-0.324	-0.741	-0.784	0.606	0.669	0.200	-0.372	0.610	-0.448
rm	-0.320	0.319	-0.417	0.062	-0.324	1.000	0.239	0.243	-0.220	-0.307	-0.351	0.147	-0.613	0.703
age	-0.668	0.573	-0.650	-0.074	-0.741	0.239	1.000	0.771	-0.467	-0.511	-0.274	0.280	-0.605	0.386
disf	-0.705	0.617	-0.737	-0.069	-0.784	0.243	0.771	1.000	-0.495	-0.546	-0.264	0.328	-0.533	0.328
rad	0.855	-0.316	0.599	-0.039	0.606	-0.220	-0.467	-0.495	1.000	0.925	0.499	-0.430	0.522	-0.409
tax	0.848	-0.323	0.710	-0.074	0.669	-0.307	-0.511	-0.546	0.925	1.000	0.491	-0.435	0.570	-0.498
ptratio	0.427	-0.382	0.391	-0.125	0.200	-0.351	-0.274	-0.264	0.499	0.491	1.000	-0.183	0.402	-0.518
black	-0.472	0.176	-0.353	0.080	-0.372	0.147	0.280	0.328	-0.430	-0.435	-0.183	1.000	-0.368	0.330
lstat	0.648	-0.412	0.628	-0.048	0.610	-0.613	-0.605	-0.533	0.522	0.570	0.402	-0.368	1.000	-0.738
medv	-0.484	0.366	-0.516	0.194	-0.448	0.703	0.386	0.328	-0.409	-0.498	-0.518	0.330	-0.738	1.000

B R script

```r
# R Script for MAS3907
# author Man Kuan Io, Lingyu Tan
# last-edited 3/4/2020

library(ElemStatLearn)
library(leaps)
library(glmnet)
library(pls)
library(nclSLR)

data(Boston, package="nclSLR")
set.seed(18053836)

sampid = sample(dim(Boston)[1], 400)

BostonNew = Boston[sampid, ]
n = nrow(BostonNew)
p = ncol(BostonNew) - 1

# reponse variable:
# logarithm of the per capita crime rate (lcrim)
y = BostonNew[, 1]
X_raw = BostonNew[, 2:14]
X = scale(X_raw)

Boston_data = data.frame(y, X)

#********************************************#
#*            Data Exploration              *#
#********************************************#
#pairs(BostonNew)
means = colMeans(BostonNew)
covariances = var(BostonNew)
correlations = cor(BostonNew)
round(correlations, 3)

#********************************************#
#*         Extra functions                  *#
#********************************************#
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)
  return(mat[,xvars] %*% coefi)
}
#********************************************#
#*         Identifying Best Model           *#
#********************************************#

#############################################
# kfolds variables                          #
# Only one kfold should be used throughout. #
#############################################
```

```
nfolds = 10
set.seed(18053836)
fold_index = sample(nfolds, n, replace=TRUE)
fold_sizes = numeric(nfolds)
fold_list = vector(mode = "list", length = 10)

for(k in 1:nfolds){
  fold_sizes[k] = length(which(fold_index==k))
}

for(i in 1:400)
{
  temp = fold_index[i]
  fold_list[[temp]] = c(fold_list[[temp]], i)
}

########################################
# Best Subset Selection (mse = 0.597) #
########################################

bss_fit = regsubsets(y ~., data=Boston_data, method="exhaustive", nvmax=p)
bss_summary = summary(bss_fit)

best_adjr2=which.max(bss_summary$adjr2)
best_cp=which.min(bss_summary$cp)
best_bic=which.min(bss_summary$bic)

par(mfrow=c(1,3))
plot(1:p, bss_summary$adjr2,xlab="Number of predictors",ylab="Adjusted Rsq",type="b")
# text(1:p, bss_summary$adjr2, labels=round(bss_summary$adjr2, 3), cex=0.7, pos=1)
points(best_adjr2, bss_summary$adjr2[best_adjr2],col="red",pch=16)
plot(1:p, bss_summary$cp,xlab="Number of predictors",ylab="Cp",type="b")
points(best_cp, bss_summary$cp[best_cp],col="red",pch=16)
plot(1:p, bss_summary$bic,xlab="Number of predictors",ylab="BIC",type="b")
points(best_bic, bss_summary$bic[best_bic],col="red",pch=16)

bss_coeff = coef(bss_fit, 6)

# print(bss_coeff)
# print(bss_summary)

# Finding best bss_fit using cross-validation
cv_bss_errors = matrix(NA, p, nfolds)
bss_mse = numeric(p)

for(k in 1:nfolds) {
  bss_tmp_fit = regsubsets(y ~ ., data=Boston_data[fold_index!=k,], method="exhaustive", nvmax=p)

  for(m in 1:p) {
```

```r
    bss_tmp_predict = predict(bss_tmp_fit, Boston_data[fold_index==k,], m)

    cv_bss_errors[m, k] = mean((Boston_data[fold_index==k,]$y - bss_tmp_predict)^2)
  }
}

for(m in 1:p) {

  bss_mse[m] = weighted.mean(cv_bss_errors[m,], w=fold_sizes)

}

best_bss_cv = which.min(bss_mse)
best_bss_mse = min(bss_mse)

print(best_bss_mse)

par(mfrow=c(1, 1))
plot(1:p, bss_mse,xlab="Number of predictors",ylab="10-fold CV Error", type="b")
points(best_bss_cv, bss_mse[best_bss_cv],col="red",pch=16)

#####################
# Ridge Regression #
#####################

grid=10^seq(5,-3,length=100)

ridge_fit=glmnet(X, y,alpha=0,standardize=FALSE,lambda=grid)
ridge_summary = summary(ridge_fit)
beta1_hat = coef(ridge_fit)

# plot for the effect of the tuning parameter by ridge regression
par(mfrow = c(1,1))

plot(ridge_fit,xvar="lambda",col=1:p,label=TRUE)

# Using cv.glmnet to find lambda_min
ridge_cv_fit=cv.glmnet(X, y, alpha=0,standardize=FALSE,lambda=grid, foldid = fold_index)

plot(ridge_cv_fit)

best_lambda = ridge_cv_fit$lambda.min
best_lambda_idx = which(ridge_cv_fit$lambda==ridge_cv_fit$lambda.min)
best_lambda_coeff = coef(ridge_fit, s = best_lambda)

##########################
# Partial Least Squares #
##########################

pls_fit = plsr(y~.,data=Boston_data,scale=FALSE)
pls_load = loadings(pls_fit)
```

```
#pls_summary = summary(pls_fit)

C = unclass(pls_load)
pls_yload = Yloadings(pls_fit)
theta1_hat = unclass(pls_yload)

pls_coeff = coef(pls_fit, intercept = T, ncomp = 2)
# round(pls_coeff, 3)

# Used to find the optimal number of components
pls_cv_fit = plsr(y ~ ., data = Boston_data, scale = FALSE, validation = "CV", segments = fold_list)

plot(pls_cv_fit,plottype="validation",legend="topright",val.type="MSEP")

pls_msep = MSEP(pls_cv_fit)
```

C Transformed Predictor Variables by PLS

```
> round(C, 5)
          Comp 1   Comp 2   Comp 3   Comp 4   Comp 5   Comp 6   Comp 7
zn      -0.25057  0.19035  0.13444 -0.52395 -0.19286  0.04276  0.43964
indus    0.34679 -0.01106 -0.25601 -0.07160 -0.35751 -0.19881  0.28084
chas    -0.01211  0.13171 -0.39714  0.40435 -0.58275  0.90878 -0.67913
nox      0.34103  0.11213 -0.22053  0.24858 -0.05535 -0.20992  0.36237
rm      -0.20047  0.60688 -0.41780 -0.06499  0.14013  0.09782 -0.05854
age     -0.31036  0.00949  0.48593 -0.22823  0.00427 -0.03584 -0.01952
disf    -0.31964 -0.06079  0.51816 -0.22823  0.01662  0.01944 -0.16528
rad      0.31709  0.40324  0.47670 -0.10090 -0.01800  0.32719  0.01109
tax      0.33887  0.31368  0.36432 -0.20391 -0.24900 -0.29760 -0.12859
ptratio  0.21555 -0.18141  0.24852 -0.51551 -0.13972  0.90684 -0.47848
black   -0.20120 -0.15831  0.16803  0.36136 -0.94222  0.25940  0.30385
lstat    0.31777 -0.29961  0.14994  0.03255  0.13810  0.21275  0.06404
medv    -0.26993  0.48667 -0.30942  0.25405 -0.03314  0.13644  0.02106
          Comp 8   Comp 9  Comp 10  Comp 11  Comp 12  Comp 13
zn      -0.34862 -0.39035  0.51559  0.00316 -0.09564  0.17487
indus    0.40964 -0.09624 -0.07640  0.00862  0.02504 -0.54017
chas     0.39525 -0.48208  0.47862 -0.16822  0.07582  0.04877
nox     -0.08082 -0.28610 -0.03699 -0.56566 -0.18271  0.35002
rm      -0.23935  0.47619 -0.17889 -0.40827  0.48080 -0.16469
age      0.43834 -0.58597 -0.15487 -0.04886  0.49283 -0.06130
disf     0.15945  0.09652  0.18381 -0.55096 -0.38553 -0.33262
rad      0.06994  0.11474  0.05897  0.12557  0.01519  0.03635
tax     -0.05397 -0.03076  0.04713  0.06265 -0.00044  0.00734
ptratio -0.04738  0.24051 -0.47722  0.04440 -0.16932  0.09241
black   -0.40882  0.44613 -0.28529  0.00102  0.12294  0.00484
lstat   -0.53953 -0.22127  0.29636 -0.15171  0.27930 -0.53569
medv    -0.13881 -0.20015 -0.24282  0.39512 -0.46125 -0.33812
attr(,"explvar")
   Comp 1    Comp 2    Comp 3    Comp 4    Comp 5    Comp 6    Comp 7
48.508037 10.007887  7.239764  6.629613  5.163153  3.216967  3.626353
   Comp 8    Comp 9   Comp 10   Comp 11   Comp 12   Comp 13
 2.689504  3.130442  5.124053  1.654223  1.459855  1.550149
```

D Contribution Statement

Each member of the group has equally contributed to the project. We both worked on the R script to ensure we agreed on one model and we edited the LaTeX document together.

Everyone should get equal marks for this project, i.e. everyone did about the same level of work.

Approved: 谭灵嵎

Lingyu Tan

Approved: ____________________

Man Kuan Io