STA 6703 SML HW6

Christopher, Madison, & Ziying

CH6. Exercise 1

1.a

The best subset selection has the smallest training RSS. This is because the best subset selects the best model from all 2^k possible models, while the forward stepwise and backward stepwise methods only iteratively add/drop one predictor based on the previous step's model without attempting all possibilities. Therefore, the best subset selection performs better on the training data set.

1.b

Any of the three selections may have the smallest test RSS. Suppose $K = 3$. If the best one-variable model is X_1 but the best two-variable contains X_2 and X_3, then the forward stepwise will fail to select the best two-variable model because the two-variable model contains X_1, resulting in a large test RSS. Similarly, in backward stepwise selection, it will fail to select the best one-variable model because the model either contains X_2 or X_3 without containing X_1. In this case, the best subset selection will have the smallest test RSS. However, if forward stepwise or backward stepwise gets the best possible model, then they will get the smallest test RSS.

1.c TRUE or FALSE

i. **TRUE** Forward selection methods forms the model with (k+1) predictors by adding an additional predictor to the model with k predictors.

ii. **TRUE** Backward selection methods forms the model with k predictors by subtracting a predictor from the model with (k+1) predictors.

iii. **FALSE** The predictors in the best (k+1)-variable model selected by the forward stepwise method may differ from the ones selected by the backward stepwise method. Thus, the predictors in the k-variable model identified by backward stepwise may not be a subset of the predictors in the (k+1)-variable model identified by forward stepwise selection.

iv. **FALSE** Similar reasons as we described in iii.

v. **FALSE** The best subset selection (k+1)-variable model by conducting all $2^{(k+1)}$ possible models, which separates from selection k-variable model.

CH6. Exercise 2

2.a Lasso

iii is correct.

The lasso performs variable selection, which forces some of the coefficient estimates to be zero by increasing λ. Thus, as the λ increases, the model becomes less flexible. As λ increases, the prediction accuracy increases when the estimated bias increases less than the decrease in variance.

2.b Ridge regression

iii is correct.

Though ridge regression cannot force some of the coefficient estimates exactly to be zero, it still aims to minimize the coefficients. Similarly, as we described in 2.a, as the λ increases, the prediction accuracy increases when the estimated bias increases less than the decrease in variance.

2.c Non-linear methods

ii is correct.

Non-linear methods aim to add some quadrative and/or cubic terms to the model to increase the model flexibility and thus decrease the estimated bias. Hence, the prediction accuracy increases when its increase in variance is less than the decrease in bias.

CH6. Exercise 4

4.a The training RSS will

steadily increase. As λ increases, the coefficient estimates tend to zero, which is similar to the number of variables decreases, and thus the training RSS will increase due to the less flexibility.

4.b The test RSS will

decrease initially, and then eventually start increasing in a U shape. As λ increases, the test RSS will decrease initially because the variance decreases larger than the increase in bias. However, when the decrease in variance is smaller than the increase in bias, the test RSS will change to increase.

4.c Variance will

steadily decrease. Similarly, as we described in 4.a, the variance decreases due to the less flexibility of the model.

4.d Squared bias will

steadily increase. Similarly, as we described in 4.a, as the λ increases, the model becomes less flexible and thus increases more noise. Hence, it causes an increase in the squared bias.

4.e The irreducible error will

remains constant. The irreducible error is random, which is irrelate to the λ.

CH6. Exercise 8

8.a

```
n=100
set.seed(0)
X <- rnorm(n)
noise <- rnorm(n)
```

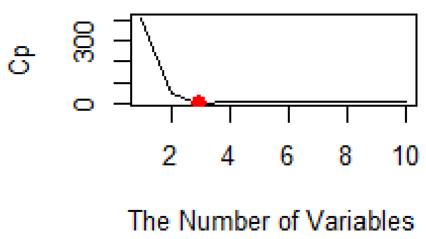
8.b

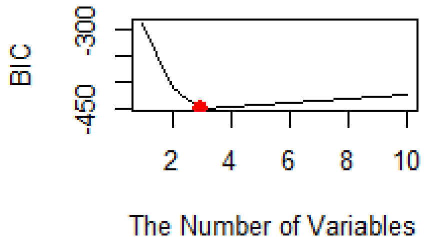
```
beta0 = 1
beta1 = 1.2
beta2 = 2
beta3 = 3

Y <- beta0 + beta1 * X + beta2 * X^2 + beta3 * X^3 + noise
```

8.c

```
#best subset selection
data <- data.frame(cbind(X,Y))
modelfit <- regsubsets(Y ~ poly(X, degree = 10, raw = T), data = data, nvmax
= 10)
results <- summary(modelfit)

which.min(results$cp)

## [1] 3

which.min(results$bic)

## [1] 3

which.max(results$adjr2)

## [1] 5

#difference in the adjusted R2
results$adjr2[which.min(results$cp)] -
results$adjr2[which.max(results$adjr2)]

## [1] -0.0001257791
```

As the plots of C_p, BIC, and adjusted R^2 show, the model with the lowest C_p and BIC is the three-variable model. The model with the maximum adjusted R^2 is the five-variable model but the difference between the three-variable model and the five-variable model is small. Therefore, we conclude that the three-variable model with X, X^2, and X^3 is the best model.





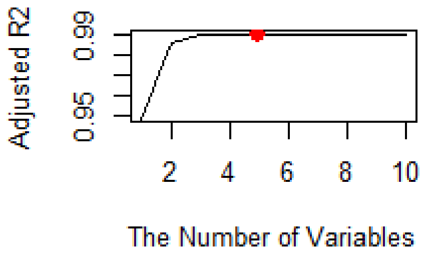


Table. The coefficients of the best model

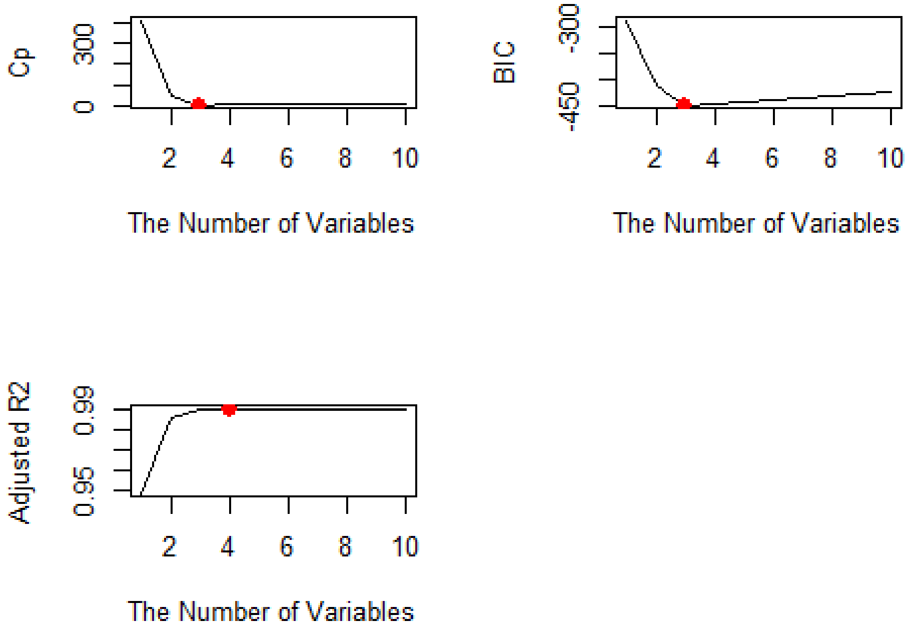
	Est. coefficient
Intercept	0.9660167
X	1.2301865
X_square	1.9706439
X_cubic	3.0503899

8.d

```
#forward stepwise selection
modelfit_fwd <- regsubsets(Y ~ poly(X, degree = 10, raw = T), data = data, 
nvmax = 10,
                           method = "forward")
results_fwd <- summary(modelfit_fwd)

#backward stepwise selection
modelfit_bwd <- regsubsets(Y ~ poly(X, degree = 10, raw = T), data = data, 
nvmax = 10,
                                method = "backward")
results_bwd <- summary(modelfit_bwd)
```

Compared to the answer in 8.c, our conclusion in the forward stepwise is the same as the best subset selection. The best model is the three-variable model with X, X^2, and X^3, which is identical to the best subset selection. However, the backward stepwise selection has different conclusion, which concludes that the best model is five-variable model with predictors X, X^2, X^5, X^7, and X^9.

Forward stepwise selection outcomes



Backward stepwise selection outcomes

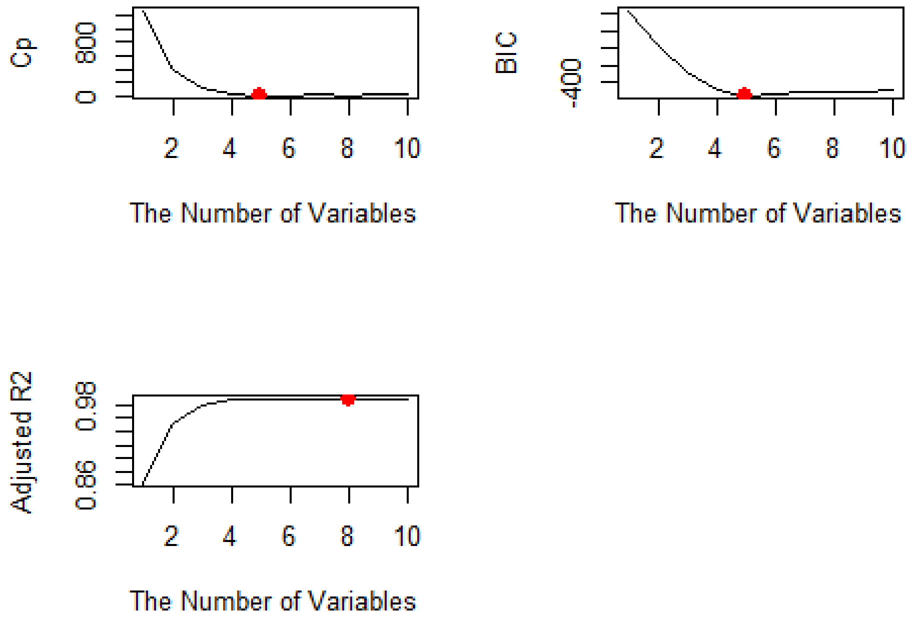


Table. The coefficients of the best model in forward stepwise selection

	Est. coefficient
Intercept	0.9660167
X	1.2301865
X_square	1.9706439
X_cubic	3.0503899

Table. The coefficients of the best model in backward stepwise selection

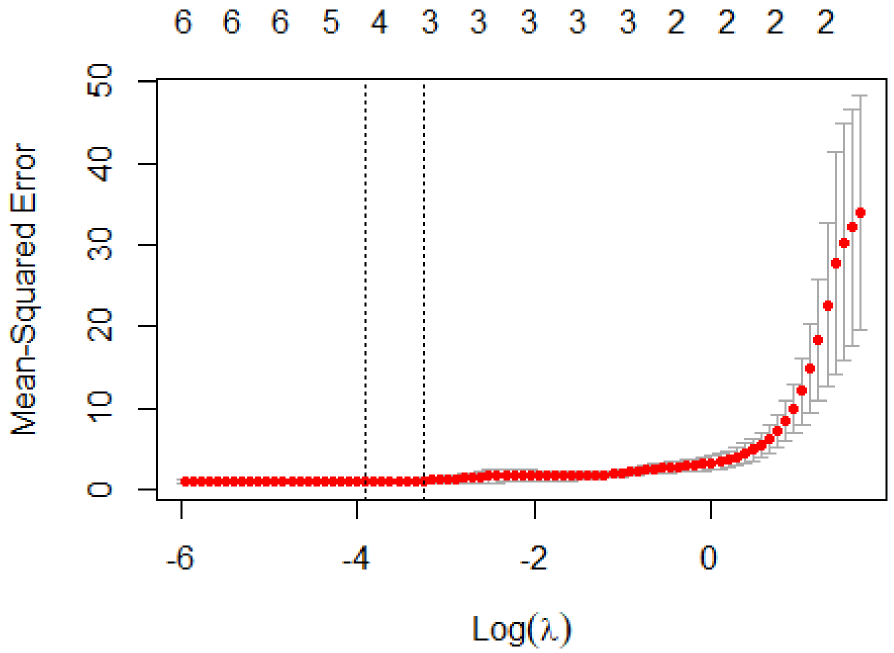
	Est. coefficient
Intercept	1.0171276
X	2.3147556
X_square	1.9052995
X_5	2.2217950
X_7	-0.5606459
X_9	0.0453176

8.e

```r
#lasso
X_matrix <- poly(X, degree = 10, raw = TRUE)
#10-fold cross-validation
set.seed(0)
```

```
train_index <- sample(1:n, size = n/2)

cv.out <- cv.glmnet(X_matrix[train_index, ], Y[train_index],
                    alpha = 1, nfolds = 10)

#test MSE is
lasso_predict <- predict(cv.out, s = cv.out$lambda.min, newx = X_matrix[-
train_index, ])

mean((lasso_predict - Y[-train_index])^2)

## [1] 1.526109
```

Plot of cross-validation errors



The optimal value of λ is

```
## [1] 0.02
```

Table. Estimated coefficients by lasso

```
modelfit_lasso <- glmnet(X_matrix, Y, alpha = 1)
results_lasso <- predict(modelfit_lasso, type = "coefficients", s =
cv.out$lambda.min)
results_lasso

## 11 x 1 sparse Matrix of class "dgCMatrix"
##                      s1
## (Intercept) 1.08421070
## 1           1.23920700
## 2           1.69850220
## 3           3.01195952
## 4           0.05602966
## 5           .
## 6           .
## 7           .
## 8           .
## 9           .
## 10          .
```

Lasso selects the best model, which is the four-variable model with predictors X, X^2, X^3, and X^4. The conclusion is different from the outcomes in the best subset, forward stepwise and backward stepwise selections.

8.f

```
#new generation
beta7 = 1.1
Y <- beta0 + beta7*X^7+noise
data2 <- data.frame(cbind(Y, X))

#the best subset selection
new_modelfit <- regsubsets(Y ~ poly(X, degree = 10, raw = TRUE), data =
data2)

new_results <- summary(new_modelfit)

which.min(new_results$cp)

## [1] 1

which.min(new_results$bic)

## [1] 1

which.max(new_results$adjr2)

## [1] 4

new_results$adjr2[which.min(new_results$cp)] -
new_results$adjr2[which.max(new_results$adjr2)]

## [1] -2.888794e-06

#The difference in the adjusted R2 between
 #the one-variable model and the four-variable model is small

#best model
t <- data.frame(coef(new_modelfit, which.min(new_results$cp)))
colnames(t) <- "Est. coefficient"
```

```
rownames(t) <- c("Intercept", "X7")
kable(t)
```

	Est. coefficient
Intercept	0.9432264
X7	1.1015262

```
#lasso
cv.out <- cv.glmnet(X_matrix[train_index,], Y[train_index], alpha = 1)
new_modelfit_lasso <- glmnet(X_matrix, Y, alpha = 1)
lasso.coef <- predict(new_modelfit_lasso, type = "coefficients", s =
cv.out$lambda.min)
lasso.coef
```

```
## 11 x 1 sparse Matrix of class "dgCMatrix"
##                      s1
## (Intercept) 1.18523374
## 1           .
## 2           .
## 3           .
## 4           .
## 5           0.06724044
## 6           .
## 7           1.05757773
## 8           .
## 9           .
## 10          .
```

For the best subset selection, the best model is the one-variable model with X^7.

For the lasso, the best model is the two-variable model with X^5 and X^7.

In this data set, the best subset selection performs better in estimates than the lasso.