Chapter 6 HW

Fabiani Rafael

Conceptual Questions

1. We perform best subset, forward stepwise, and backward stepwise selection on a single data set. For each approach, we obtain $p+1$ models, containing $0, 1, 2, \ldots, p$predictors. Explain your answers:

(a) Which of the three models with k predictors has the smallest *training* RSS?

- The model with k predictors that has the smallest training RSS is the one obtained by use of the best subset selection method. This is due to how best subset selection considers all possible combinations of predictors and selects the one that minimizes the training RSS.

(b) Which of the three models with k predictors has the smallest test RSS?

- The model with k predictors that has the smallest test RSS is not necessarily the one obtained by best subset selection. It could be any of the three methods, depending on how well they generalize to unseen data. The test RSS is influenced by the model's ability to generalize, which is not guaranteed to be the best subset selection method. Moreover the best subset selection method may overfit the training data and consequently yielding a higher test RSS. Now for forward and backward stepwise selection, theres no gurantee that the model with k predictors will be the one that minimizes the test RSS.

(c) True or False:

i. The predictors in the k-variable model identified by forward stepwise are a subset of the predictors in the $(k+1)$ variable model identified by forward stepwise selection.

- True, because forward stepwise selection adds predictors one at a time, and the k-variable model will always be a subset of the $(k+1)$ variable model.

ii. The predictors in the k-variable model identified by backward stepwise are a subset of the predictors in the $(k+1)$ variable model identified by backward stepwise selection.

- True. Backward stepwise selection removes predictors one at a time, so the k-variable model will always be a subset of the $(k+1)$ variable model.

iii. The predictors in the k-variable model identified by backward stepwise are a subset of the predictors in the (k + 1) variable model identified by forward stepwise selection.

- False, because backward stepwise selection removes predictors, while forward stepwise selection adds predictors. The k-variable model from backward stepwise selection may not be a subset of the $(k+1)$ variable model from forward stepwise selection.

iv. The predictors in the k-variable model identified by forward stepwise are a subset of the predictors in the $(k+1)$ variable model identified by backward stepwise selection.

- False, since forward stepwise selection adds predictors, while backward stepwise selection removes predictors. So the k-variable model from forward stepwise selection might not be a subset of the $(k+1)$ variable model from backward stepwise selection.

v. The predictors in the k-variable model identified by best subset are a subset of the predictors in the $(k+1)$ variable model identified by best subset selection.

- False, best subset selection considers all possible combinations of predictors and does not guarantee that the k-variable model will be a subset of the $(k+1)$ variable model so it may be that the k-variable model may include different predictors than those in the $(k+1)$.

For parts (a) through (c), indicate which of i. through iv. is correct. Justify your answer.

(a) The lasso, relative to least squares, is:

i. More flexible and hence will give improved prediction accuracy when its increase in bias is less than its decrease in variance.

ii. More flexible and hence will give improved prediction accuracy when its increase in variance is less than its decrease in bias.

iii. Less flexible and hence will give improved prediction accuracy when its increase in bias is less than its decrease in variance.

iv. Less flexible and hence will give improved prediction accuracy when its increase in variance is less than its decrease in bias.

- Of these iii is correct. The optimal selection for λ will result in a decrease in the variance of the model, which will lead to an increase in bias. This is because the lasso shrinks the coefficients towards zero, which reduces the model's flexibility and variance.

(b) Repeat (a) for ridge regression relative to least squares.

- iii is correct. The optimal selection for λ will result in a decrease in the variance of the model, which will lead to an increase in bias. This is because ridge regression shrinks the coefficients towards zero, which reduces the model's flexibility and variance.

(c) Repeat (a) for non-linear methods relative to least squares.

- Of these ii is correct. A non linear method is more flexible than least squares, and it can improve prediction accuracy when the increase in variance is less than the decrease in bias. A possible advantage of non-linear methods is their improved potential to capture complex relationships in the data and yield better predictions.

Applied Questions

9. In this exercise, we will predict the number of applications received using the other variables in the College data set.

(a) Split the data set into a training set and a test set.

```r
set.seed(448)
train = sample(1:nrow(College), nrow(College)/2)
test = -train
College.test = College[test, ]
College.train = College[train, ]

# dimensions of the training && test sets
dim(College.train)
```

```
## [1] 388  18
```

```r
dim(College.test)
```

```
## [1] 389  18
```

(b) Fit a linear model using least squares on the training set, and report the test error obtained.

```r
lm.fit = lm(Apps ~ ., data = College.train)
lm.pred = predict(lm.fit, newdata = College.test)
lm.mse = mean((College.test$Apps - lm.pred)^2)
lm.mse
```

```
## [1] 1570431
```

(c) Fit a ridge regression model on the training set, with λ chosen by cross-validation. Report the test error obtained.

```r
x.train = model.matrix(Apps ~ ., data = College.train)[, -1]
y.train = College.train$Apps
x.test = model.matrix(Apps ~ ., data = College.test)[, -1]
y.test = College.test$Apps
set.seed(448)
grid = 10^seq(10, -2, length = 100)
ridge.mod = glmnet(x.train, y.train, alpha = 0, lambda = grid)
cv.out = cv.glmnet(x.train, y.train, alpha = 0)
bestlam = cv.out$lambda.min
ridge.pred = predict(ridge.mod, s = bestlam, newx = x.test)
ridge.mse = mean((y.test - ridge.pred)^2)
ridge.mse
```

```
## [1] 2318655
```

(d) Fit a lasso model on the training set, with λ chosen by cross validation. Report the test error obtained, along with the number of non-zero coefficient estimates.

```r
set.seed(448)

# lasso with CV
lasso.cv <- cv.glmnet(x.train, y.train, alpha = 1)
bestlam.lasso <- lasso.cv$lambda.min

# tst MSE
lasso.pred <- predict(lasso.cv, s = bestlam.lasso, newx = x.test)
lasso.mse <- mean((College.test$Apps - lasso.pred)^2)
lasso.mse
```

```
## [1] 1569760
```

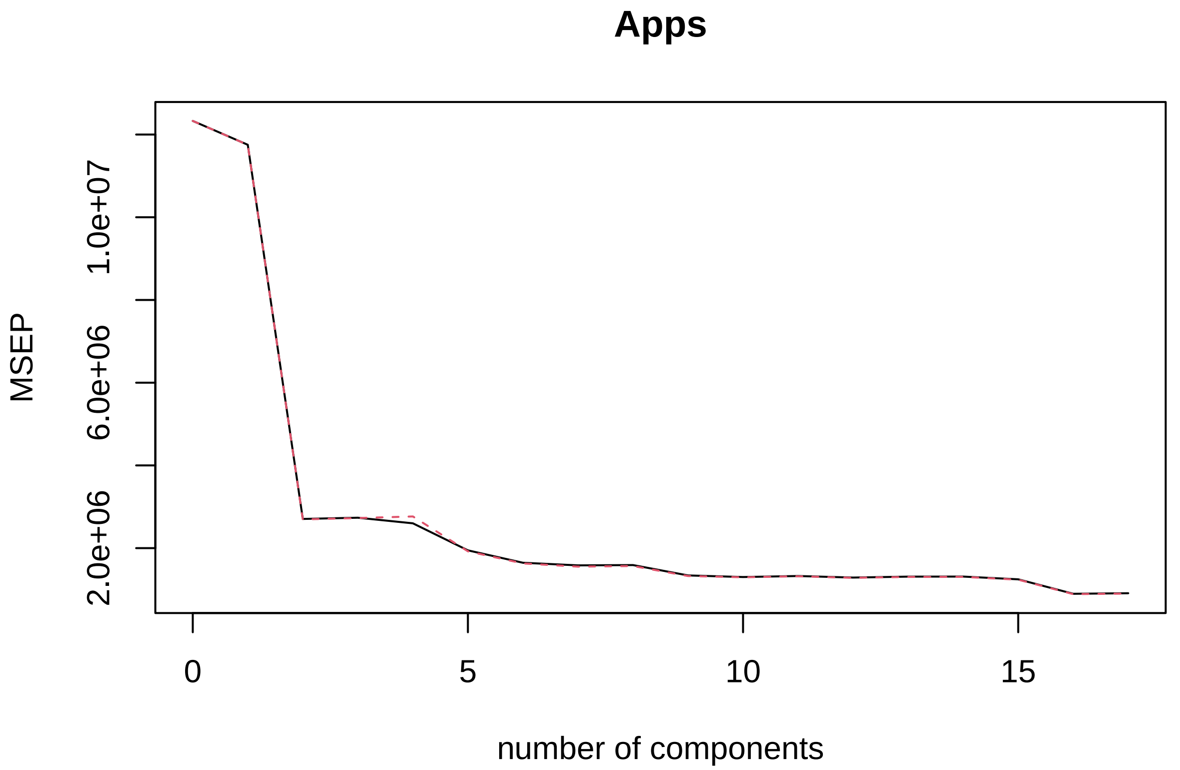
```r
# num  non-zero coefficients

lasso.coef <- coef(lasso.cv, s = bestlam.lasso)
# exclud intercept
num.nonzero <- sum(lasso.coef[-1, ] != 0)
num.nonzero
```

```
## [1] 15
```

(e) Fit a PCR model on the training set, with M chosen by cross validation. Report the test error obtained, along with the value of M selected by cross-validation.

```r
set.seed(448)
# PCR with CV
pcr.fit <- pcr(Apps ~ ., data = College.train, scale = TRUE, validation = "CV")
validationplot(pcr.fit, val.type = "MSEP")
```



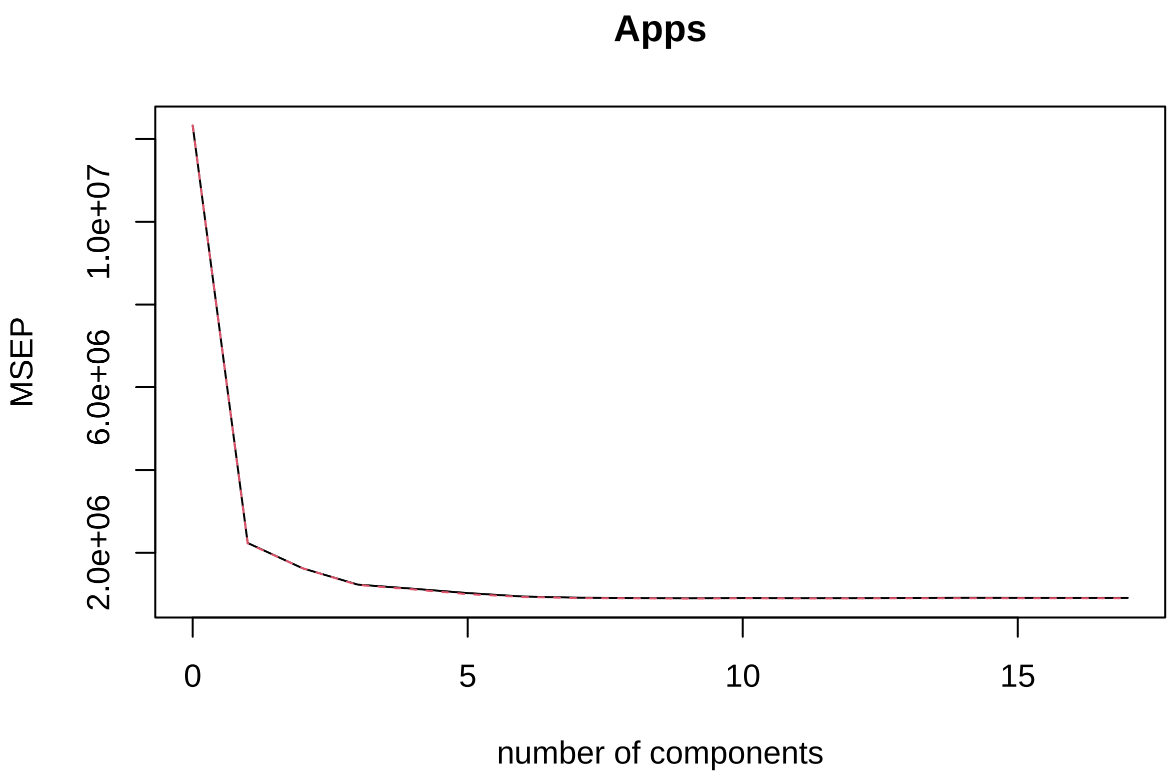
```
# best M , M: = 16s
pcr.bestM <- which.min(pcr.fit$validation$PRESS)

# tst MSE
pcr.pred <- predict(pcr.fit, College.test, ncomp = pcr.bestM)
pcr.mse <- mean((College.test$Apps - pcr.pred)^2)
pcr.mse
```

```
## [1] 1688927
```

(f) Fit a PLS model on the training set, with M chosen by cross validation. Report the test error obtained, along with the value of M selected by cross-validation.

```
set.seed(448)
# pls with cv
pls.fit <- plsr(Apps ~ ., data = College.train, scale = TRUE, validation = "CV")
validationplot(pls.fit, val.type = "MSEP")
```



```r
# best M , M: = 16
pls.bestM <- which.min(pls.fit$validation$PRESS)

# tst MSE
pls.pred <- predict(pls.fit, College.test, ncomp = pls.bestM)
pls.mse <- mean((College.test$Apps - pls.pred)^2)
pls.mse
```

```
## [1] 1602263
```

(g) Comment on the results obtained. How accurately can we predict the number of college applications received? Is there much difference among the test errors resulting from these five approaches?

- Linear 1 570 431
- Ridge 2 318 655
- Lasso 1 569 760 (15 non-zero β)
- PCR 1 688 927 (M = 16)
- PLS 1 602 263 (M = 16)
- The linear model and lasso yielded the lowest test error, indicating that they are the most accurate in predicting the number of college applications received. Ridge regression showed the highest error, while PCR and PLS were intermediate and very similar.