Stats216v: Statistical Learning

Stanford University
Summer 2017
Gyu-Ho Lee (gyuhox@gmail.com (mailto:gyuhox@gmail.com))

6. Linear Model Selection and Regularization

6.1.R1

Which of the following modeling techniques performs Feature Selection?

1. Linear Discriminant Analysis
2. Least Squares
3. Linear Regression with Forward Selection
4. Support Vector Machines

Gyu-Ho's Answer: 3.

Forward Selection chooses a subset of the predictor variables for the final model. The other three methods end up using all of the predictor variables.

6.2.R1

We perform best subset and forward stepwise selection on a single dataset. For both approaches, we obtain $p + 1$ models, containing $0, 1, 2, \dots, p$ predictors.

Which of the two models with k predictors is guaranteed to have training RSS no larger than the other model?

1. Best Subset
2. Forward Stepwise
3. They always have the same training RSS
4. Not enough information is given to know

Gyu-Ho's Answer: 1.

Best subset selection may have the smallest test RSS because it takes into account more models than the other methods. However, the other methods might also pick a model with smaller test RSS by sheer luck..

6.2.R2

Which of the two models with k predictors has the smallest test RSS?

1. Best Subset
2. Forward Stepwise
3. They always have the same test RSS
4. Not enough information is given to know

Gyu-Ho's Answer: 4.

We know that Best Subset selection will always have the lowest training RSS (that is how it is defined). That said, we don't know which model will perform better on a test set.

6.3.R1

You are trying to fit a model and are given $p = 30$ predictor variables to choose from. Ultimately, you want your model to be interpretable, so you decide to use Best Subset Selection.

How many different models will you end up considering?

Gyu-Ho's Answer: 2^{30}.

Each predictor can either be included or not included in the model. That means that for each of the 30 variables there are two options. Thus, there are 2^{30} potential models.

Note: Don’t ever try to fit that many models! It is too many and that is why Best Subset Selection is rarely used in practice for say p=10 or larger.

6.3.R2

How many would you fit using Forward Selection?

Gyu-Ho's Answer: 466.

For Forward Selection, you fit $(p - k)$ models for each $k = 0, \ldots p - 1$. The expression for the total number of models fit: $1 + \frac{p(p+1)}{2} = 1 + \frac{30*31}{2}$.

6.4.R1

You are fitting a linear model to data assumed to have Gaussian errors. The model has up to $p = 5$ predictors and $n = 100$ observations. Which of the following is most likely true of the relationship between C_p and AIC in terms of using the statistic to select a number of predictors to include?

1. C_p will select a model with more predictors AIC.
2. C_p will select a model with fewer predictors AIC.
3. C_p will select the same model as AIC.
4. Not enough information is given to decide.

Gyu-Ho's Answer: 4.

3.

For **linear models with Gaussian errors**, Cp and AIC and equivalent.

6.5.R1

You are doing a simulation in order to compare the effect of using Cross-Validation or a Validation set. For each iteration of the simulation, you generate new data and then use both Cross-Validation and a Validation set in order to determine the optimal number of predictors. Which of the following is most likely?

1. The Cross-Validation method will result in a higher variance of optimal number of predictors.
2. The Validation set method will result in a higher variance of optimal number of predictors.
3. Both methods will produce results with the same variance of optimal number of predictors.
4. Not enough information is given to decide.

Gyu-Ho's Answer: 2.

Cross-Validation is similar to doing a Validation set multiple times and then averaging the answers. As such, we expect it to have lower variance than the Validation set method. This is why Cross-Validation is appealing (especially for small n).

6.6.R1

$\sqrt{\sum_p^{j=1} \beta_j^2}$ is equivalent to:

Gyu-Ho's Answer: L2 norm of β.

$\sqrt{\sum_p^{j=1} \beta_j^2} = ||\beta||^2$

6.6.R2

You perform ridge regression on a problem where your third predictor, x_3, is measured in dollars. You decide to refit the model after changing x_3 to be measured in cents. Which of the following is true?:

1. $\hat{\beta}_3$ and $\hat{y}$ will remain the same.
2. $\hat{\beta}_3$ will change but $\hat{y}$ will remain the same.
3. $\hat{\beta}_3$ will remain the same but $\hat{y}$ will change.
4. $\hat{\beta}_3$ and $\hat{y}$ will both change.

Gyu-Ho's Answer: 1.

4.

The units of the predictors affects the L2 penalty in ridge regression, and hence $\hat{\beta}_3$ and $\hat{y}$ will both change

6.7.R1

Which of the following is NOT a benefit of the sparsity imposed by the Lasso?

1. Sparse models are generally more easy to interperet.
2. The Lasso does variable selection by default.
3. Using the Lasso penalty helps to decrease the bias of the fits.
4. Using the Lasso penalty helps to decrease the variance of the fits.

Gyu-Ho's Answer: 3.

Restricting ourselves to simpler models by including a Lasso penalty will generally decrease the variance of the fits at the cost of higher bias.

6.8.R1

Which of the following would be the worst metric to use to select λ in the Lasso?

1. Cross-Validated error
2. Validation set error
3. RSS

Gyu-Ho's Answer: 3.

RSS would be the worst metric to use because it will cause us to always select the most complicated model. Any of the other metrics could be used, although Cross-Validated error is probably most common.

6.9.R1

We compute the principal components of our p predictor variables. The RSS in a simple linear regression of Y onto the largest principal component will always be no larger than the RSS in a simple regression of Y onto the second largest principal component. True or False? (You may want to watch 6.10 as well before answering - sorry!)

Gyu-Ho's Answer: False.

Adding more variables reduces the Residual Square Sums (RSS) in a linear model.

The answer is simply that we are using the least squares method. Any set of coefficients we choose must give a sum of squared residuals at least as great as for the best fit. Suppose we fit the model with the coefficients of the additional variables set to zero. This is the same as the fit without the additional variables, and as it restricts the coefficients, the sum of squares must be suboptimal except in the unlikely event that the least squares fit has these coefficinetns exactly zero.

Principal components are found independently of Y, so we can't know the relationship with Y a priori.

6.10.R1

You are working on a regression problem with many variables, so you decide to do Principal Components Analysis first and then fit the regression to the first 2 principal components. Which of the following would you expect to happen?:

1. A subset of the features will be selected.
2. Model Bias will decrease relative to the full least squares model.
3. Variance of fitted values will decrease relative to the full least squares model.
4. Model interpretability will improve relative to the full least squares model.

Gyu-Ho's Answer: 3.

While some forms of dimensional reduction will cause the first or fourth to occur, that is not the case with PCA. When using dimensional reduction we restrict ourselves to simpler models. Thus, we expect bias to increase and variance to decrease.

6.Q.1

Suppose we estimate the regression coefficients in a linear regression model by minimizing

$$\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij}) + \lambda \sum_{j=1}^{p} \beta_j^2$$

for a particular value of λ. For each of the following, select the correct answer:

- As we increase λ from 0, the **training RSS** will:

Gyu-Ho's Answer: Steadily increase.

Increasing λ will force us to fit simpler models. This means that training RSS will steadily increase because we are less able to fit the training data exactly.

- As we increase λ from 0, the **test RSS** will:

Gyu-Ho's Answer: Decrease initially, and then eventually start increasing in a U shape.

At first, we expect test RSS to improve because we are not overfitting our training data anymore. Eventually, we will start fitting models that are too simple to capture the true effects and test RSS will go up.

- As we increase λ from 0, the **variance** will:

Gyu-Ho's Answer: Steadily decrease.

Increasing λ will cause us to fit simpler models, which reduces the variance of the fits.

- As we increase λ from 0, the **(squared) bias** will:

Gyu-Ho's Answer: Steadily increase.

Increasing λ will cause us to fit simpler models, which have larger squared bias.

- As we increase λ from 0, the **irreducible error** will:

Gyu-Ho's Answer: Remain constant.

Increasing λ will have no effect on irreducible error. By definition, irreducible error is an aspect of the problem and has nothing to do with a particular model being fit.

6.Q.1-1

Suppose we estimate the regression coefficients in a linear regression model by minimizing

$\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij})$ subject to $\sum_{j=1}^{p} |\beta_j| \leq s$

for a particular value of λ. For each of the following, select the correct answer:

- As we increase λ from 0, the **training RSS** will:

Gyu-Ho's Answer: Steadily increase.

- As we increase λ from 0, the **test RSS** will:

Gyu-Ho's Answer: Decrease initially, and then eventually start increasing in a U shape.

- As we increase λ from 0, the **variance** will:

Gyu-Ho's Answer: Steadily decrease.

- As we increase λ from 0, the **(squared) bias** will:

Gyu-Ho's Answer: Steadily increase.

- As we increase λ from 0, the **irreducible error** will:

Gyu-Ho's Answer: Remain constant.

6.R.R1

One of the functions in the glmnet package is cv.glmnet(). This function, like many functions in R, will return a list object that contains various outputs of interest. What is the name of the component that contains a vector of the mean cross-validated errors?

In [1]:
```
LoadLibraries = function() {
    library(MASS)
    install.packages("ISLR")
    library(ISLR)
    install.packages("leaps")
    library(leaps)
    install.packages("pls")
    library(pls)
    print("Libraries have been loaded!")
}

LoadLibraries()
```

```
Updating HTML index of packages in '.Library'
Making 'packages.html' ... done
Updating HTML index of packages in '.Library'
Making 'packages.html' ... done
Updating HTML index of packages in '.Library'
Making 'packages.html' ... done

Attaching package: 'pls'

The following object is masked from 'package:stats':

    loadings


[1] "Libraries have been loaded!"
```

In [2]:
```
names(Hitters)
dim(Hitters)
Hitters = na.omit(Hitters)
dim(Hitters)
```

'AtBat' 'Hits' 'HmRun' 'Runs' 'RBI' 'Walks' 'Years' 'CAtBat' 'CHits' 'CHmRun' 'CRuns' 'CRBI' 'CWalks' 'League' 'Division' 'PutOuts' 'Assists' 'Errors' 'Salary' 'NewLeague'

322 20

263 20

In [3]:

```
library(glmnet)

# model.matrix to produce a matrix with 19 predictors
# also automatically transforms any qualitative variables into dummy var
iables
x = model.matrix(Salary~., Hitters)[,-1]
y = Hitters$Salary
grid = 10^seq(10, -2, length=100)

# alpha=0 for ridge regression
# alpha=1 for lasso
# automatically standardize variables
ridge.mod = glmnet(x, y, alpha=0, lambda=grid)
dim(coef(ridge.mod))
```

```
Loading required package: Matrix
Loading required package: foreach
Loaded glmnet 2.0-5
```

20 100

In [4]:

```
# split samples into training set and test set
# to estimate test error of ridge regression, lasso
set.seed(1)
train = sample(1:nrow(x), nrow(x)/2)
test = (-train)
y.test = y[test]
```

In [7]:

```
# use cross-validation to choose λ
set.seed(1)
cv.out = cv.glmnet(x[train,], y[train], alpha=0)
names(cv.out)
```

'lambda' 'cvm' 'cvsd' 'cvup' 'cvlo' 'nzero' 'name' 'glmnet.fit'
'lambda.min' 'lambda.1se'

In [10]:
```
# lambda.min is the value of λ that gives minimum mean cross-validated er
ror
cv.out$lambda.min

# contains a vector of the mean cross-validated errors
cv.out$cvm
```

211.741584781282

214354.303637251	213164.708864405	212292.015886432	212085.979574303
211861.028528945	211615.551030778	211347.819509483	211055.989690819
210738.098514156	210392.063370548	210015.682995141	209606.640393825
209162.508230901	208680.758274873	208158.757947734	207593.705024591
206982.972523307	206323.835151519	205613.444938001	204849.0620957
204028.029452135	203147.823047691	202206.109641969	201200.810935912
200130.173987882	198992.846902772	197787.958409088	196515.199421301
195174.904134216	193768.127643148	192296.716569609	190763.36874804
189171.6777536	187526.153569562	185832.258691162	184096.387312978
182325.62228014	180527.736955726	178711.408641509	176885.679248302
175059.90957603	173243.589909034	171446.139190335	169676.701409249
167943.948460026	166255.898693723	164619.813209113	163042.062502039
161527.575079549	160080.311091669	158703.366985309	157399.333113961
156169.139851723	155012.915367284	153930.109807678	152919.53912513
151979.517997744	151107.992056028	150302.663700497	149561.168026721
148880.594649595	148258.567388932	147695.308013481	147186.411913869
146727.433461154	146320.375377957	145961.813983003	145648.394285883
145378.467172561	145155.58023841	144974.873035186	144831.441079251
144726.027706479	144654.900147812	144617.044677928	144606.301830691
144623.530938551	144664.407824805	144728.059578698	144808.98990538
144907.142629995	145016.549259666	145135.626680521	145265.863387186
145397.120813286	145534.406552462	145675.893264413	145811.569975284
145946.620051613	146078.020812919	146206.383507617	146327.728292901
146442.204136978	146550.1860599	146649.689588485	146741.094410638
146824.430621866	146899.156499292		