HW1 Report for STAT 542

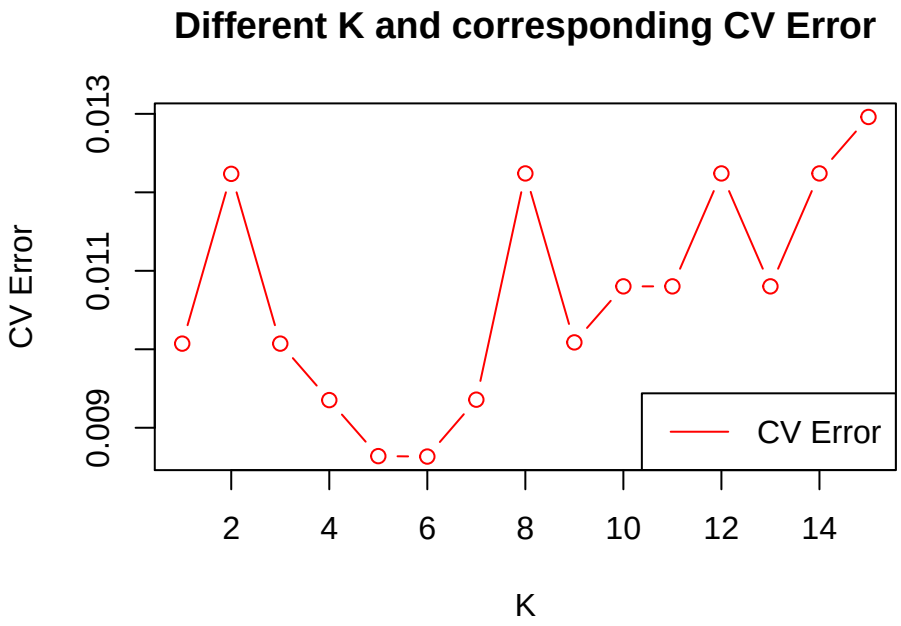
Wenke Huang, September 9, 2016

Question 1

1.a Cross-validation fitting

Here we use the 10-fold Cross-Validation to find the best K value based on training dataset only. After calculating, the overall result is shown below numerically and as a plot.

```
##                  [,1]       [,2]       [,3]        [,4]        [,5]
## CVError.1a 0.01007202 0.0122355 0.01007202 0.009352592 0.008638381
##                  [,6]        [,7]       [,8]       [,9]      [,10]
## CVError.1a 0.008633168 0.009357806 0.01224072 0.01008766 0.01080187
##                 [,11]      [,12]      [,13]      [,14]      [,15]
## CVError.1a 0.01080187 0.01224072 0.01080187 0.01224072 0.01296014
```



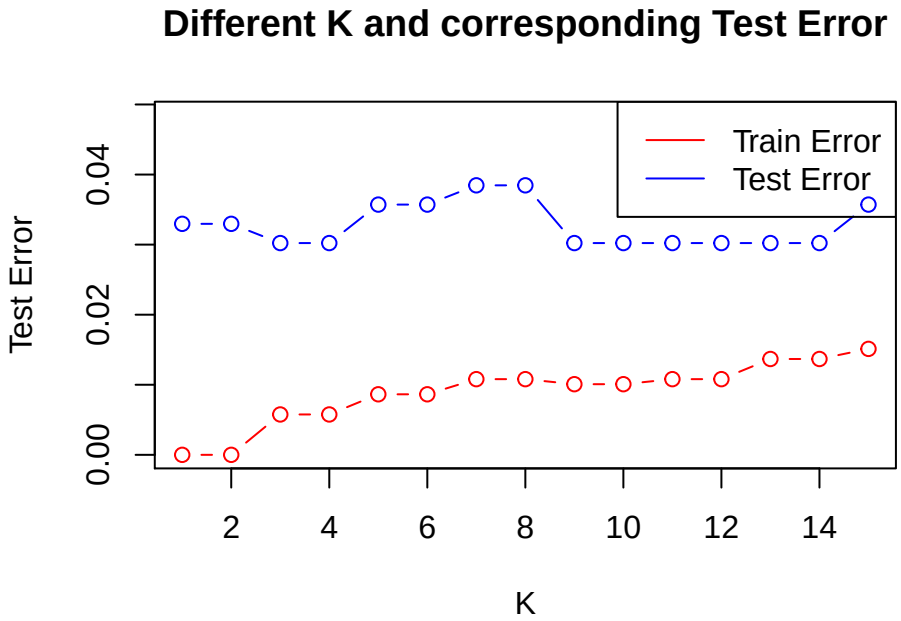
We will choose the K with the smallest CV Error. So the best K value we can get in this situation is 6 and its Cross-Validation Error is 0.0086332.

1.b Normal method fitting

Here we use the training dataset to fit different Ks and use testing dataset to do the test and find the best K. After calculating, the overall result is shown below numerically and as a plot.

```
##                [,1] [,2]        [,3]        [,4]        [,5]        [,6]
## TrainError.1b    0    0 0.005759539 0.005759539 0.008639309 0.008639309
##                      [,7]       [,8]       [,9]      [,10]      [,11]
## TrainError.1b 0.01079914 0.01079914 0.01007919 0.01007919 0.01079914
##                     [,12]      [,13]      [,14]      [,15]
## TrainError.1b 0.01079914 0.01367891 0.01367891 0.01511879
```

```
##                     [,1]       [,2]       [,3]       [,4]       [,5]
## TestError.1b 0.03296703 0.03296703 0.03021978 0.03021978 0.03571429
##                     [,6]       [,7]       [,8]       [,9]      [,10]
## TestError.1b 0.03571429 0.03846154 0.03846154 0.03021978 0.03021978
##                    [,11]      [,12]      [,13]      [,14]      [,15]
## TestError.1b 0.03021978 0.03021978 0.03021978 0.03021978 0.03571429
```



We will choose the K with the smallest test error. If test errors of different Ks are the same, we will choose the largest K, the model of which will have a greatest degrees of freedom than the others. So the best K value we can get in this situation is 3 and its Test Error is 0.0302198.

1.c Comparation and conclusion

When we use Cross-Validation, we will fit and test 10 times and calculate their mean value each time we choose a different K. More times of fitting and testing will definitely make the output more reliable. However, if we use the second method, we only fit one model and then test its error each time we choose a different K, which is obviously less dependable and easily affected by particular points.

Model	CV (K=6)	Normal method (K=3)
Test Error	0.032967	0.0302198

We can also test the first model by using the test dataset like the second model. Even though we can find that the test error of first model is a little larger than the second one, I think this difference could be allowed to exist.

In conclusion, using Cross-Validation to select the best K value is my preference. In this case, I will choose 6 as my best K value and use it for further analysis.

Question 2

2.a Derive degrees of freedom

Based on the definition of k-nearest neighbor, we can find that covariance between $\hat{y}_i$ and y_i should be n/k. Because only the nearest point will be considered as its predicted value while the rest points are all noncorrelated.

degrees of freedom $= \sum_{i=1}^{n} Cov(\hat{y}_i, y_i)/\sigma^2 = \sum_{i=1}^{n} \frac{\sigma^2}{k} \times \frac{1}{\sigma^2} = \sum_{i-1}^{n} \frac{1}{k} = \frac{n}{k}$

In this case, we know that k = 5. So degrees of freedom will be $n/5$.

2.b Generate features X

In this and following 2 parts, we will only output the first 5 rows of each matrix or vector that we generate because the data are too large. The whole data will still be saved in the original R code. Part of X (200*4) we generate is shown below.

```
##           X1          X2           X3         X4
## 1  1.5524573  0.06571647  1.196574938  0.6247744
## 2  1.6675547 -1.20225214 -0.007901432  0.6760396
## 3 -0.1345583 -0.43070455 -0.836625661 -0.4564392
## 4 -0.1889437 -0.48894507  1.602155128  0.3172683
## 5 -0.8581125  0.97392105  1.609068176 -1.3266976
```

2.c Define the mean of Y

We would like to define the mean of Y as $E(Y) = f(X) = \frac{1}{4} \times (X_1 + X_2 + X_3 + X_4)$. Part of $f(X)$ value will be shown below.

```
##          [,1]      [,2]       [,3]      [,4]       [,5]
## E_Y 0.8598808 0.2833602 -0.4645819 0.3103837 0.09954479
```

2.d Generate response Y and then predict

As required, we generate the response Y. The formula is $Y = E(Y) + \epsilon$ while ϵ is an independent standard normal noise $N(0, 1)$. Then we use them to fit a 5-nearest neighbor model, and calculate fitted response values $\hat{Y}$. Part of $\hat{Y}$ will be shown below.

```
##                  [,1]     [,2]       [,3]        [,4]       [,5]
## Y_Predicted 2.123446 1.685559 -0.5443119 0.007172874 -0.5935385
```

2.e Repeat previous part and calculate degrees of freedom

We repeat the previous part 10 times, and calculate the corvariance between y_i and $\hat{y}_i$. We also know that the response's variance σ^2 is 1 which is exactly the same with the variance of ϵ. So, we can get that degrees of freedom should be 37.5074482 based on the original formula provided by `2.a`.

2.f Comparation and conclusion

Repeat Times	10	50	100	Theoretical DF
Degrees of Freedom	37.5074482	41.0388493	40.2191197	40

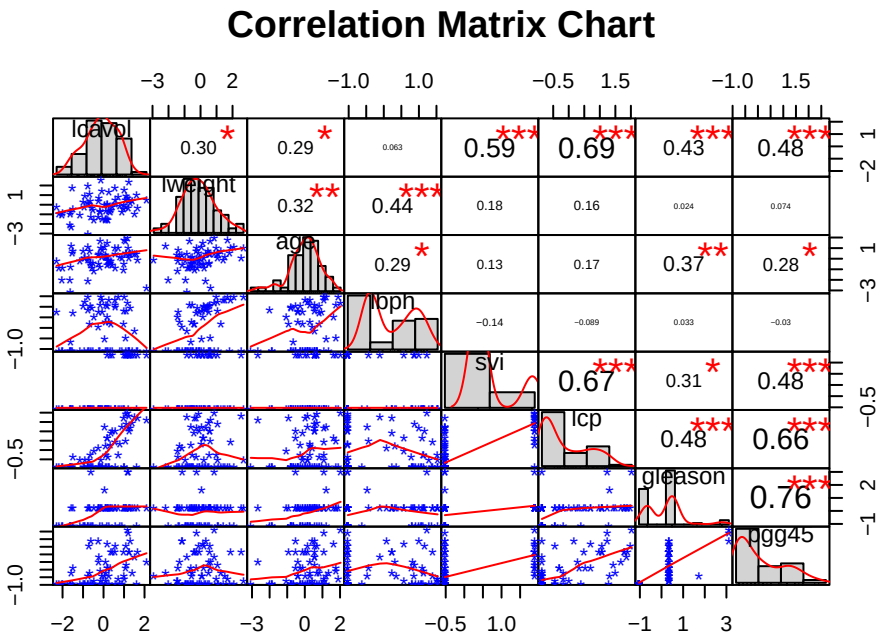
The definition of degrees of freedom has been provided in `2.a`. Based on that formula and data, we can get that the degrees of freedom in this case should be 37.5074482, which is not far from the theoretical degrees of freedom 40 (We can get it from 200/5).

In this part, we also tried to repeat generating response values 50 and 100 times and then calculate their degrees of freedom by using the original formula. All results calculated in `Question 2` are shown in the above table. Based on it and basic theory of statistics, we can know that if we can repeat more and more, the degrees of freedom we will get should be closer and closer to the theoretical one.

Question 3

3.1 Linear regression with all predictors

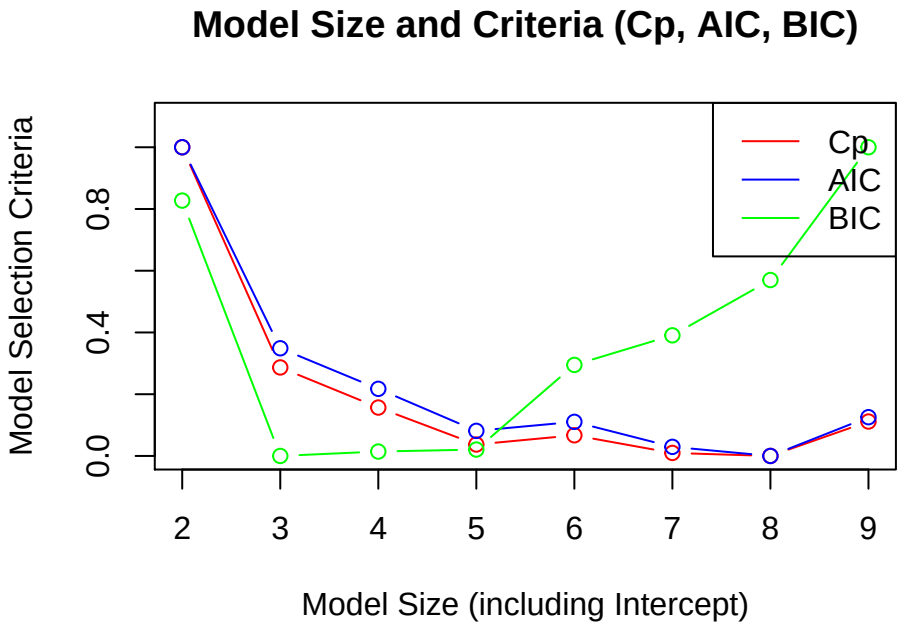
In this case, variable `lpsa` is the response while standardlized value of rest variables are all predictors. First, we visualize all of predictors of training dataset. We can clearly find that many predictors are strongly correlated with others, such as `lcp` and `svi`, `gleason` and `pgg45`, which is definitely not a good sign. We should search a model which does not contain all of predictors in case of over-fitting. So we fit a linear model with all of the 8 predictors. The coefficients of this model are shown below.



Linear model with all predictors

```
## (Intercept)      lcavol     lweight         age        lbph         svi
##  2.46493292  0.67952814  0.26305307 -0.14146483  0.21014656  0.30520060
##         lcp     gleason       pgg45
## -0.28849277 -0.02130504  0.26695576
```

3.2 Backward stepwise regresssion and subsets



We use the backward stepwise regression to look for the its subsets. The maximum size of subsets should be set as 8 so we will not ignore any possibly better results. In order to make

the results more clear and readable, We use modified Cp, AIC and BIC which are scaled to [0,1] as model selection criteria here. The plot shown above tells us that when best model sizes (including Intercept) will be 8 when we use Cp or AIC and 3 when we choose BIC.

Only knowing the model size is far less than being enough, we will retrieve and extract the predictors and their coefficients of the best models generated by different criteria. These procedures will make our conclusion more quantitively reliable and dependable .

Model genereted by `Cp`

```
## (Intercept)      lcavol     lweight         age        lbph         svi
##   2.4668675   0.6764486   0.2652760  -0.1450300   0.2095349   0.3070936
##         lcp       pgg45
##  -0.2872242   0.2522850
```

Model genereted by `AIC`

```
## (Intercept)      lcavol     lweight         age        lbph         svi
##   2.4668675   0.6764486   0.2652760  -0.1450300   0.2095349   0.3070936
##         lcp       pgg45
##  -0.2872242   0.2522850
```

Model genereted by `BIC`

```
## (Intercept)      lcavol     lweight
##   2.4773573   0.7397137   0.3163282
```

We can also find that the result generated by Cp and AIC are exactly the same.

3.3 Comparation and conclusion

Criteria	Cp	AIC	BIC	Linear Regression
Train MSE	0.4393627	0.4393627	0.5536096	0.4391998
Test MSE	0.5165135	0.5165135	0.4924823	0.521274

We calculate train and test MSEs of the subsets and full model. We can find out that the test MSE of model `FITBIC` is the smallest from the above table, which should be considered the best under this circumstance. In this situation, I suggest that we choose model `BICFit` while its predictors and coefficients will be following.

```
## (Intercept)      lcavol     lweight
##   2.4773573   0.7397137   0.3163282
```