STAA 577 HW2

Enqun Wang

April 1, 2016

Due: April 4 at noon

A major issue with antiretroviral drugs is the mutation of the virus' genes. Because of its high rate of replication (109 to 1010 virus per person per day) and error-prone polymerase1, HIV can easily develop mutations that alter susceptibility to antiretroviral drugs. The emergence of resistance to one or more antiretroviral drugs is one of the more common reasons for therapeutic failure in the treatment of HIV.

In the paper, a sample of in vitro HIV viruses were grown and exposed to a particular antiretroviral therapy. The susceptibility of the virus to treatment and the number of genetic mutations of each virus were recorded.

1. Load the data set `hiv.rda` and create

```
load(file = "/Users/ewenwang/Dropbox/CSU-MAS/STAA 577/Homework/hiv.rda")

X = hiv.train$x
Y = hiv.train$y

str(hiv.train)
```

```
## List of 2
##  $ x: num [1:704, 1:208] 1 0 0 0 0 0 0 0 0 0 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr [1:208] "p1" "p2" "p3" "p4" ...
##  $ y: num [1:704] 2.301 0.633 0.623 0.544 2.301 ...
```

What would be n and p in this problem? What are the features in this problem? What are the observations? What is the response (i.e. supervisor)? **Note:** Attempt to answer this question before moving on to the rest of the questions.

- For training data, $n = 704, p = 208$.
- Four mutation sets were used as inputfeatures: a complete set of all mutations $\geq$ 2 sequences in thedata set, the 30 most common data set mutations, an expert panelmutation set, and a set of nonpolymorphic treatment-selectedmutations from a public database linking protease and reversetranscriptase sequences to antiretroviral drug exposure.
- The susceptibility of the virus to treatment is the supervisor.

---

2. Consider the design matrix $\mathbb{X}$. It is composed of 0's and 1's, with a 1 indicating a mutation in a particular gene. Run

```
table(X)
```

```
## X
##      0      1
## 135589  10843
```

What results do you get? What does this indicate?

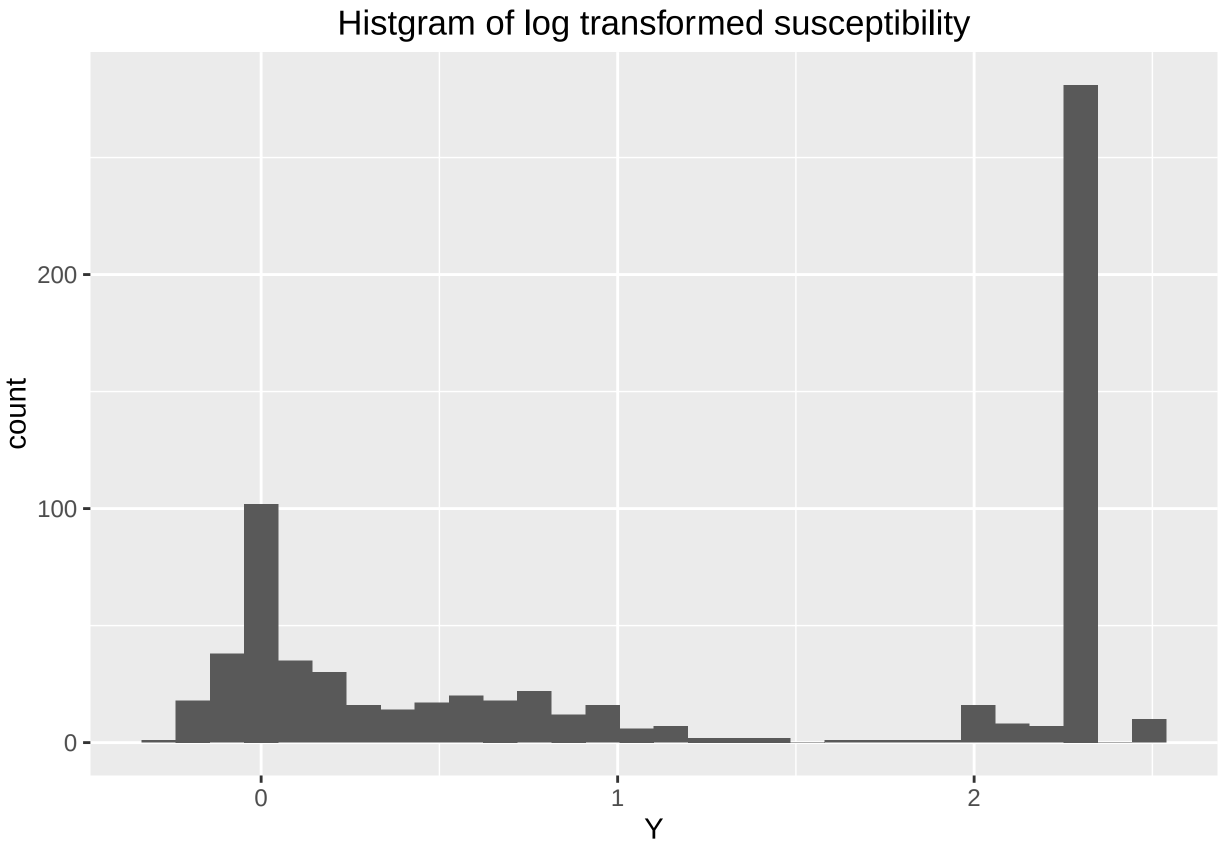
- For 704 virus, each with 208 features, there are totally 10843 genetic mutations and 135589 non genetic mutations.

---

3. The response is the log transformed susceptibility of a virus to the considered treatment, with large values indicating the virus is resistant (that is, not susceptible). Run

```r
require(ggplot2)

qplot(Y, geom = "histogram", bins = 30, main = "Histgram of log transformed susceptibility")
```



What plot did you just create? What does this indicate?

- We created a histgram of log transformed susceptibility. From the plot, it is clear that two groups of data distributed around 0 and 2.3, the data is bimodally distributed. The plot indicates that a large amount of virus are resistant; while there are still numbers of virus are susceptible.

---

4. We may have (at least) two goals with a data set such as this: inference or prediction. An inferential question would be: can we find some genes whose mutation seems to be most related to viral susceptibility?

A prediction question would be: can we make a model that would predict whether this therapy would be efficacious, given a virus with a set of genetic mutations.

(a) Let's do model selection, which can address either of these goals.

i. Try to find the best subset solution for this problem. Discuss any problems or findings you discover. In particular, how many possible models are there?

- There are 4.1137614×10^{62} possible models when we take all $p = 208$ into account with best subset selection. However, there are dependency exit in the X matrix. It is better to remove linearly dependent columns, then find the best subset solution.

```
rankifremoved <- sapply(1:ncol(X), function(x)qr(X[,-x])$rank)
X.removed <- X[, which(rankifremoved == max(rankifremoved))]

df <- data.frame(Y, X = X.removed)
colnames(X.removed)
```

```
##  [1] "p12"  "p13"  "p14"  "p17"  "p18"  "p26"  "p29"  "p30"  "p76"  "p89"
## [11] "p209" "p217" "p220" "p222" "p229" "p231"
```

- This suggests that only 16 variables are linearly independent. And we try to build models based on these variables.

```
require(leaps)

regfit.full=regsubsets(Y~., data=df, nvmax = length(which(rankifremoved == max(rankifremoved))))
```

```
## Warning in leaps.setup(x, y, wt = wt, nbest = nbest, nvmax = nvmax,
## force.in = force.in, : 12 linear dependencies found
```

```
## Reordering variables and trying again:
```

```
reg.summary <- summary (regfit.full)
reg.summary
```

```
## Subset selection object
## Call: regsubsets.formula(Y ~ ., data = df, nvmax = length(which(rankifremoved ==
##     max(rankifremoved))))
## 16 Variables  (and intercept)
##        Forced in Forced out
## X.p12      FALSE      FALSE
## X.p26      FALSE      FALSE
## X.p76      FALSE      FALSE
## X.p209     FALSE      FALSE
## X.p13      FALSE      FALSE
## X.p14      FALSE      FALSE
## X.p17      FALSE      FALSE
## X.p18      FALSE      FALSE
## X.p29      FALSE      FALSE
## X.p30      FALSE      FALSE
## X.p89      FALSE      FALSE
## X.p217     FALSE      FALSE
## X.p220     FALSE      FALSE
## X.p222     FALSE      FALSE
## X.p229     FALSE      FALSE
## X.p231     FALSE      FALSE
## 1 subsets of each size up to 4
## Selection Algorithm: exhaustive
##          X.p12 X.p13 X.p14 X.p17 X.p18 X.p26 X.p29 X.p30 X.p76 X.p89
## 1  ( 1 ) "*"   " "   " "   " "   " "   " "   " "   " "   " "   " "  
## 2  ( 1 ) "*"   " "   " "   " "   " "   " "   " "   " "   " "   " "  
## 3  ( 1 ) "*"   " "   " "   " "   " "   " "   " "   " "   " "   "*"  
## 4  ( 1 ) "*"   " "   " "   " "   " "   "*"   " "   " "   "*"   " "  
##          X.p209 X.p217 X.p220 X.p222 X.p229 X.p231
## 1  ( 1 ) " "    " "    " "    " "    " "    " "   
## 2  ( 1 ) " "    " "    " "    " "    " "    "*"   
## 3  ( 1 ) " "    " "    " "    " "    " "    "*"   
## 4  ( 1 ) "*"    " "    " "    " "    " "    " "   
```

```r
which.min(reg.summary$bic)
```

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

```r
which.min(reg.summary$cp)
```

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

```r
which.max(reg.summary$adjr2)
```

```
## [1] 2
```

- Setting `nvmax = 16`, there are 2517 possible models.
- Both BIC and Cp suggest the model containing p_{12}, while $R_{adjusted}$ suggests the model with p_{12} and p_{231}.

ii. Now do forward selection with `regsubsets`. Report the selected covariates using `bic` as the criterion.

```r
regfit.full = regsubsets(Y ~ ., df, method ="forward", nvmax =16)
```

```
## Warning in leaps.setup(x, y, wt = wt, nbest = nbest, nvmax = nvmax,
## force.in = force.in, : 12 linear dependencies found
```

```
## Reordering variables and trying again:
```

```r
reg.summary <- summary (regfit.full)
reg.summary
```

```
## Subset selection object
## Call: regsubsets.formula(Y ~ ., df, method = "forward", nvmax = 16)
## 16 Variables  (and intercept)
##        Forced in Forced out
## X.p12      FALSE      FALSE
## X.p26      FALSE      FALSE
## X.p76      FALSE      FALSE
## X.p209     FALSE      FALSE
## X.p13      FALSE      FALSE
## X.p14      FALSE      FALSE
## X.p17      FALSE      FALSE
## X.p18      FALSE      FALSE
## X.p29      FALSE      FALSE
## X.p30      FALSE      FALSE
## X.p89      FALSE      FALSE
## X.p217     FALSE      FALSE
## X.p220     FALSE      FALSE
## X.p222     FALSE      FALSE
## X.p229     FALSE      FALSE
## X.p231     FALSE      FALSE
## 1 subsets of each size up to 4
## Selection Algorithm: forward
##          X.p12 X.p13 X.p14 X.p17 X.p18 X.p26 X.p29 X.p30 X.p76 X.p89
## 1  ( 1 ) "*"   " "   " "   " "   " "   " "   " "   " "   " "   " "  
## 2  ( 1 ) "*"   " "   " "   " "   " "   " "   " "   " "   " "   " "  
## 3  ( 1 ) "*"   " "   " "   " "   " "   " "   " "   " "   "*"   " "  
## 4  ( 1 ) "*"   " "   " "   " "   " "   "*"   " "   " "   "*"   " "  
##          X.p209 X.p217 X.p220 X.p222 X.p229 X.p231
## 1  ( 1 ) " "    " "    " "    " "    " "    " "   
## 2  ( 1 ) "*"    " "    " "    " "    " "    " "   
## 3  ( 1 ) "*"    " "    " "    " "    " "    " "   
## 4  ( 1 ) "*"    " "    " "    " "    " "    " "   
```

```r
reg.summary$bic
```

```
## [1]  4.092115  9.599298 15.272598 21.558077
```

- Taking BIC as criterion, we select p_{12} as predictor.

(b) Now, let's do ridge regression, which only addresses prediction. Using the package `glmnet`, plot the CV curve over the grid of λ values and indicate the minimum, and finally report the CV estimate of the prediction risk for $\hat{\beta}_{ridge,\hat{\lambda}}$.

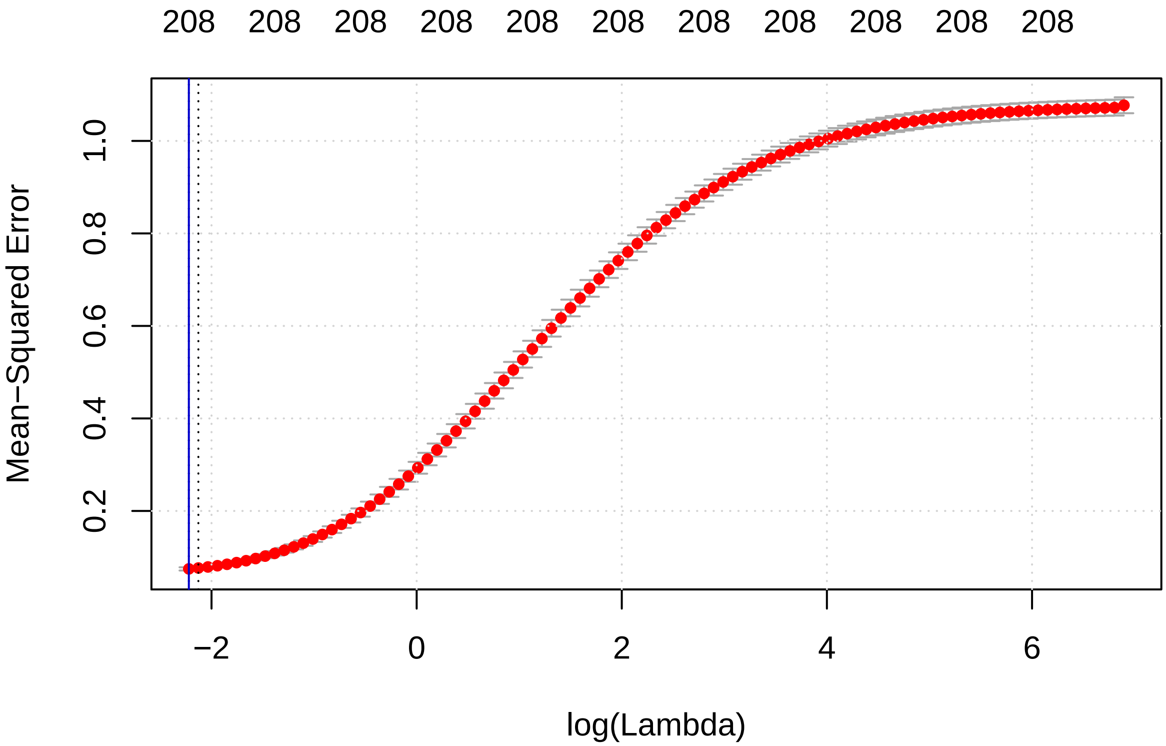
Note: There is no need to report the p coefficient estimates from the ridge solution.

```r
require(glmnet)

grid = 10^seq (10,-2, length =100)
ridge.mod = glmnet(X,Y,alpha = 0, lambda = grid)

set.seed (1)
cv.out = cv.glmnet (X,Y,alpha =0)
bestlam = cv.out$lambda.min
```

```r
plot(cv.out)
abline(v = log(bestlam), col = "blue3")
grid()
```



```r
ridge.pred = predict(ridge.mod, s=bestlam, newx=hiv.test$x)
risk.pred = mean((ridge.pred -hiv.test$y)^2)
```

- The CV estimate of the prediction risk for $\hat{\beta}_{ridge,\hat{\lambda}} = 0.1023393$.