CompBio HW3

Fei Xia

Lasso

Loading data:

```
library(glmnet)
```

```
## Loading required package: Matrix
## Loading required package: foreach
## Loaded glmnet 2.0-2
```

```
library(lars)
```

```
## Loaded lars 1.2
```

```
dataset<-read.table("prostate.txt")
x <- as.matrix(dataset[,1:8])
y <- as.matrix(dataset[,9])
n=length(dataset[,1])
```

Results of glment:

```
reg_res <- cv.glmnet(x, y, nfolds=5)
parameters <- coef(reg_res$glmnet.fit, s=reg_res$lambda.1se)
print(parameters)
```

```
## 9 x 1 sparse Matrix of class "dgCMatrix"
##                       1
## (Intercept) 0.5173355
## lcavol      0.4616486
## lweight     0.3451527
## age         .
## lbph        .
## svi         0.3939218
## lcp         .
## gleason     .
## pgg45       .
```

My Implementation of Lasso:

```
# normalization
X <- scale(x)
x_coef <- attr(X,"scaled:scale")
Y <- y-mean(y)

# lasso optimization
```

```
lambda <- reg_res$lambda.1se
eps = 1e-8
A <- t(X)%*%X
b <- t(X)%*%Y
for (i in 1:8)
  {
    A[i,i]=A[i,i]+lambda
  }
w=solve(A)%*%b

for (step in 1:100)
  {
    w0 <- w
    for (i in 1:8)
      {
        c <- t(X[,i])%*%(Y-X%*%w+w[i]*X[,i])/n
        w[i] <- sign(c)*max((abs(c)-lambda),0)
      }
    if(max(abs(w-w0))<eps)
      {
        break
      }
  }
w=w/x_coef
print(w)
```

```
##              [,1]
## lcavol  0.4557090
## lweight 0.3435818
## age     0.0000000
## lbph    0.0000000
## svi     0.3969634
## lcp     0.0000000
## gleason 0.0000000
## pgg45   0.0000000
```