stat790 a2

2023-02-12

```
library(matlib)
```

```
## Warning in rgl.init(initValue, onlyNULL): RGL: unable to open X11 display
## Warning: 'rgl.init' failed, running with 'rgl.useNULL = TRUE'.
```

1a)

Naive linear algebra:

Suppose X is a full column rank matrix and y is the vector contains all response variable values, we have: $\vec{\beta} = (X'X)^{-1}X'\vec{y}$

QR decomposition:

suppose X could be decomposed to an orthogonal matrix Q and an upper triangular matrix R, $\vec{\beta} = R^{-1}Q'\vec{y}$

SVD:

Suppose X could be decomposed as the product of U, Σ, and V', where U and V are orthogonal matriics andd Σ is diagonal matrix with the singular values of X
$\vec{\beta} = V\Sigma^{-1}U'\vec{y}$

Cholesky decomposition:

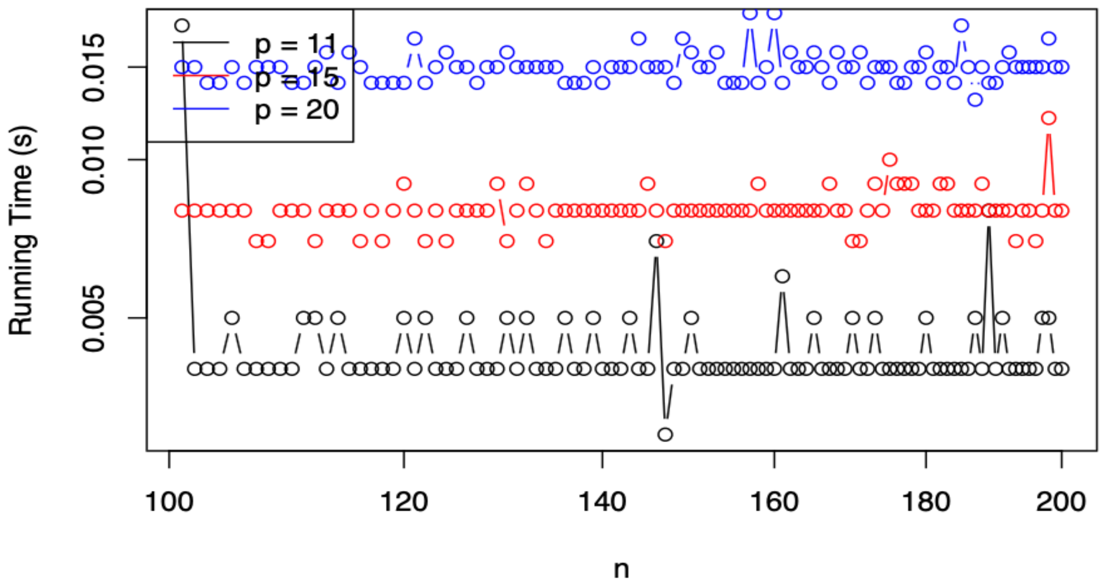
Suppose X could be decomposed as the product of L and L', L is a lower triangular matrix.
$\vec{\beta} = (L')^{-1}L^{-1}X'\vec{y}$

1b

```
naive1=function(n,p){
  s.star=1
  z1= c(rep(1, n/2), rep(0, n/2))
  Z = matrix(rnorm(n * (p - 2)), nrow = n, ncol = (p - 2))
  X = cbind(1, z1, Z)
  b.star = p^(-1) * rnorm(p)#randomly select
  y = X %*% b.star + s.star * rnorm(n)
  inv(t(X) %*% X) %*% t(X) %*% y
  }

n=200
p=20
timing <- matrix(NA, nrow = n-100, ncol = p-10)
for (i in 101:n){
  for(j in 11:p){
    timing[i-100, j-10] <- system.time(naive1(i,j))["elapsed"]
    }
  }
```

```
plot(101:n, timing[,1], type = "b", log = "xy", xlab = "n", ylab = "Running Time (s)")
lines(101:n, timing[,5], type = "b", col = "red")
lines(101:n, timing[,10], type = "b", col = "blue")
legend("topleft", legend = c("p = 11", "p = 15", "p = 20"), col = c("black", "red", "blue"), lty = 1)
```



we could see that when p goes higher, the running time is higher, when n goes higher, the running time will remain unchange.

```
ps=11:p
ns=101:200
lm(log(colMeans(timing))~log(ps))
```

```
##
## Call:
## lm(formula = log(colMeans(timing)) ~ log(ps))
##
## Coefficients:
## (Intercept)      log(ps)
##     -10.396        2.062
```

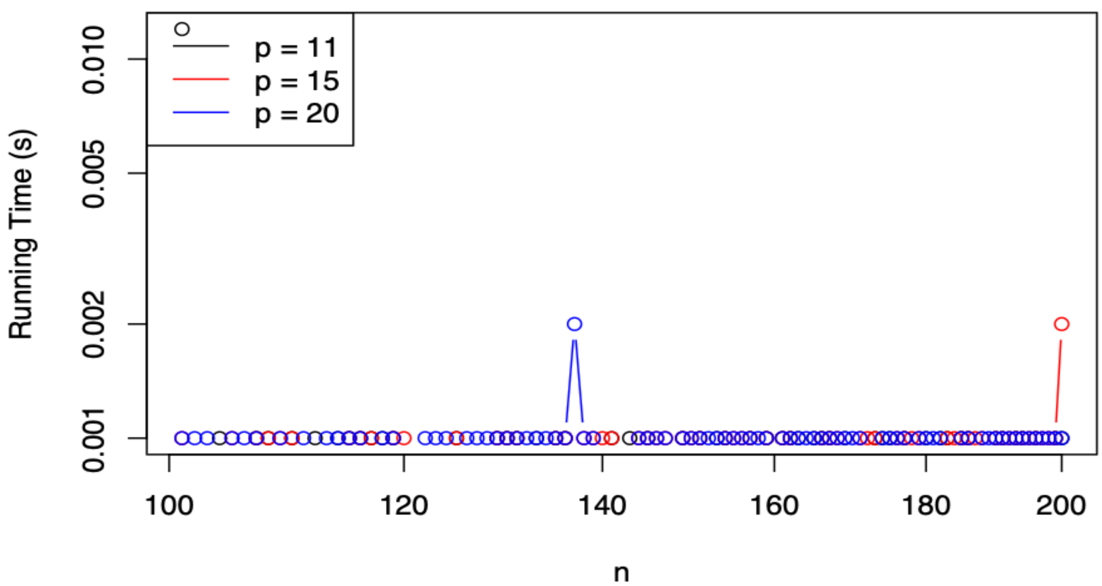
```
lm(log(rowMeans(timing))~log(ns))
```

```
##
## Call:
## lm(formula = log(rowMeans(timing)) ~ log(ns))
##
## Coefficients:
## (Intercept)      log(ns)
##    -4.89694      0.03812
```

from the coefficients of the linear regression model, we could see that when p increase 1 percent, the running time will increase 2 percent, one percent change in n will only result 0.04 percent increase in running time. which matched our computational results.

```r
set.seed(1)
QR=function(n,p){
  s.star=1
  z1= c(rep(1, n/2), rep(0, n/2))
  Z = matrix(rnorm(n * (p - 2)), nrow = n, ncol = (p - 2))
  X = cbind(1, z1, Z)
  out = qr(x = X)
  b.star = p^(-1) * rnorm(p)#randomly select
  y = X %*% b.star + s.star * rnorm(n)
  qr.coef(qr(x = X), y = y)
}
```

```r
n=200
p=20
timing1 <- matrix(NA, nrow = n-100, ncol = p-10)
for (i in 101:n){
  for(j in 11:p){
    timing1[i-100, j-10] <- system.time(QR(i,j))["elapsed"]
    }
}
plot(101:n, timing1[,1], type = "b", log = "xy", xlab = "n", ylab = "Running Time (s)")
lines(101:n, timing1[,5], type = "b", col = "red")
lines(101:n, timing1[,10], type = "b", col = "blue")
legend("topleft", legend = c("p = 11", "p = 15", "p = 20"), col = c("black", "red", "blue"), lty = 1)
```




```r
ps=c(11:p)
ns=101:200
lm(log(colMeans(timing1))~log(ps))
```

```
##
## Call:
## lm(formula = log(colMeans(timing1)) ~ log(ps))
##
```

```
## Coefficients:
## (Intercept)      log(ps)
##     -8.9850       0.6066
```

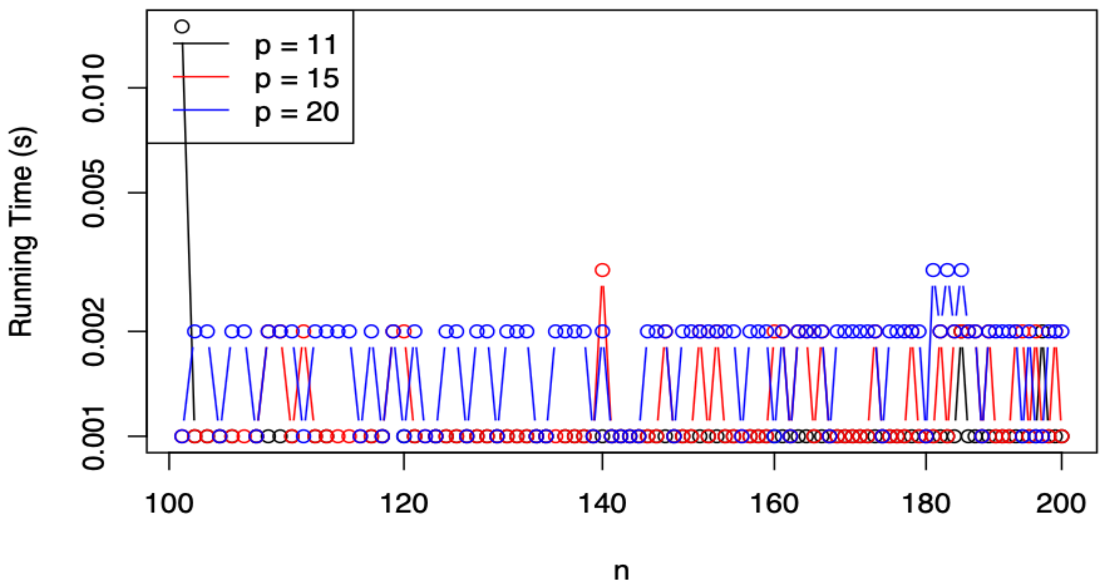
```
lm(log(rowMeans(timing1))~log(ns))
```

```
##
## Call:
## lm(formula = log(rowMeans(timing1)) ~ log(ns))
##
## Coefficients:
## (Intercept)      log(ns)
##     -9.7725       0.4802
```

from the computational results, we could find that higher p and higher n will leads to higher running time, but it is not significant, which matches the results of regression model.

```
set.seed(1)
svd1=function(n,p){
  s.star=1
  z1 = c(rep(1, n/2), rep(0, n/2))
  Z = matrix(rnorm(n * (p - 2)), nrow = n, ncol = (p - 2))
  X = cbind(1, z1, Z)
  b.star = p^(-1) * rnorm(p)#randomly select
  y = X %*% b.star + s.star * rnorm(n)
  svd(X)$v %*% diag(1 / svd(X)$d) %*% t(svd(X)$u) %*% y
}
```

```
n=200
p=20
timing2 <- matrix(NA, nrow = n-100, ncol = p-10)
for (i in 101:n){
  for(j in 11:p){
    timing2[i-100, j-10] <- system.time(svd1(i,j))["elapsed"]
    }
}
plot(101:n, timing2[,1], type = "b", log = "xy", xlab = "n", ylab = "Running Time (s)")
lines(101:n, timing2[,5], type = "b", col = "red")
lines(101:n, timing2[,10], type = "b", col = "blue")
legend("topleft", legend = c("p = 11", "p = 15", "p = 20"), col = c("black", "red", "blue"), lty = 1)
```




```
lm(log(colMeans(timing2))~log(ps))
```

```
##
## Call:
## lm(formula = log(colMeans(timing2)) ~ log(ps))
##
## Coefficients:
## (Intercept)      log(ps)
##     -9.1601       0.9266
```

```
lm(log(rowMeans(timing2))~log(ns))
```

```
##
## Call:
## lm(formula = log(rowMeans(timing2)) ~ log(ns))
##
## Coefficients:
## (Intercept)      log(ns)
##     -8.0695       0.2877
```

we could find on the plot that higher p will has a higher running time, haigher n will also increase the running time, but not significant, which matches the results of regression models.

2

```
data_link <- "https://hastie.su.domains/ElemStatLearn/datasets/prostate.data"
prostate <- read.table(data_link, header = TRUE)


set.seed(1)
indice <- sample(1:nrow(prostate), nrow(prostate) * 0.7)
traindata <- prostate[indice, ]
testdata <- prostate[-indice, ]


ridge.augment <- function(x, y, lambda) {
  n <- nrow(x)
  p <- ncol(x)
  X_t <- rbind(x, sqrt(lambda) * diag(p))
  y_t <- c(y, rep(0, p))
  beta_t <- solve(t(X_t) %*% X_t, t(X_t) %*% y_t)
  beta_t[1:p]

}


x_train <- as.matrix(traindata[, -c(1, 2, 9)])
y_train <- traindata[, 9]
lambdas <- seq(0, 1, length.out = 100)
start_time1 <- Sys.time()
betas <- matrix(NA, nrow = length(lambdas), ncol = ncol(x_train))
for (i in seq_along(lambdas)) {
  betas[i, ] <- ridge.augment(x_train, y_train, lambdas[i])
}
end_time1 <- Sys.time()
total_time1 <- end_time1 - start_time1

# Fit a native ridge regression model using the glmnet package
library(glmnet)
```

```
## Loading required package: Matrix
```

```
## Loaded glmnet 4.1-6
```

```
start_time2 <- Sys.time()
x_train_scaled <- scale(x_train)
y_train_scaled <- scale(y_train)
glmnet_fit <- glmnet(x_train_scaled, y_train_scaled, alpha = 0, lambda = lambdas)
end_time2 <- Sys.time()
total_time2 <- end_time2 - start_time2
t(betas)
```

```
##              [,1]         [,2]         [,3]         [,4]         [,5]
## [1,]  0.013226062  0.013244255  0.013262388  0.013280463  0.013298480
## [2,]  0.099203572  0.099068467  0.098933829  0.098799655  0.098665943
## [3,]  1.157267006  1.155240478  1.153221439  1.151209848  1.149205662
## [4,]  0.198325345  0.198661308  0.198995843  0.199328957  0.199660659
## [5,]  0.230973144  0.230861356  0.230749869  0.230638681  0.230527792
## [6,] -0.001401301 -0.001394322 -0.001387361 -0.001380418 -0.001373494
```

```
## [7,]  0.008353186  0.008264883  0.008176905  0.008089251  0.008001918
##               [,6]         [,7]         [,8]         [,9]        [,10]
## [1,]  0.013316439  0.013334341  0.013352185  0.013369973  0.013387704
## [2,]  0.098532690  0.098399894  0.098267553  0.098135663  0.098004223
## [3,]  1.147208842  1.145219347  1.143237135  1.141262168  1.139294405
## [4,]  0.199990957  0.200319857  0.200647369  0.200973500  0.201298257
## [5,]  0.230417199  0.230306902  0.230196897  0.230087184  0.229977762
## [6,] -0.001366587 -0.001359698 -0.001352826 -0.001345972 -0.001339135
## [7,]  0.007914904  0.007828209  0.007741830  0.007655765  0.007570014
##              [,11]        [,12]       [,13]        [,14]        [,15]
## [1,]  0.013405378  0.013422997  0.01344056  0.013458067  0.013475519
## [2,]  0.097873230  0.097742681  0.09761257  0.097482908  0.097353679
## [3,]  1.137333807  1.135380334  1.13343395  1.131494611  1.129562283
## [4,]  0.201621648  0.201943681  0.20226436  0.202583701  0.202901703
## [5,]  0.229868628  0.229759782  0.22965122  0.229542945  0.229434951
## [6,] -0.001332316 -0.001325514 -0.00131873 -0.001311962 -0.001305212
## [7,]  0.007484573  0.007399443  0.00731462  0.007230104  0.007145892
##              [,16]        [,17]        [,18]        [,19]        [,20]
## [1,]  0.013492917  0.013510260  0.013527549  0.013544784  0.013561965
## [2,]  0.097224884  0.097096523  0.096968591  0.096841088  0.096714010
## [3,]  1.127636926  1.125718504  1.123806977  1.121902310  1.120004465
## [4,]  0.203218376  0.203533728  0.203847766  0.204160496  0.204471927
## [5,]  0.229327239  0.229219807  0.229112653  0.229005776  0.228899175
## [6,] -0.001298478 -0.001291761 -0.001285061 -0.001278377 -0.001271711
## [7,]  0.007061984  0.006978377  0.006895070  0.006812061  0.006729350
##              [,21]        [,22]        [,23]        [,24]        [,25]
## [1,]  0.013579093  0.013596167  0.013613189  0.013630159  0.013647076
## [2,]  0.096587357  0.096461124  0.096335310  0.096209913  0.096084931
## [3,]  1.118113405  1.116229094  1.114351495  1.112480574  1.110616293
## [4,]  0.204782065  0.205090916  0.205398489  0.205704790  0.206009826
## [5,]  0.228792848  0.228686794  0.228581011  0.228475499  0.228370255
## [6,] -0.001265060 -0.001258426 -0.001251809 -0.001245207 -0.001238622
## [7,]  0.006646933  0.006564811  0.006482980  0.006401441  0.006320190
##              [,26]        [,27]        [,28]        [,29]        [,30]
## [1,]  0.013663941  0.013680754  0.013697516  0.013714227  0.013730886
## [2,]  0.095960361  0.095836201  0.095712450  0.095589104  0.095466162
## [3,]  1.108758618  1.106907513  1.105062944  1.103224875  1.101393273
## [4,]  0.206313603  0.206616129  0.206917410  0.207217453  0.207516264
## [5,]  0.228265278  0.228160567  0.228056121  0.227951938  0.227848017
## [6,] -0.001232053 -0.001225499 -0.001218962 -0.001212440 -0.001205935
## [7,]  0.006239228  0.006158551  0.006078159  0.005998051  0.005918225
##              [,31]        [,32]        [,33]        [,34]        [,35]
## [1,]  0.013747495  0.013764054  0.013780563  0.013797021  0.013813430
## [2,]  0.095343622  0.095221481  0.095099738  0.094978390  0.094857435
## [3,]  1.099568103  1.097749331  1.095936924  1.094130847  1.092331069
## [4,]  0.207813850  0.208110218  0.208405374  0.208699325  0.208992076
## [5,]  0.227744356  0.227640955  0.227537812  0.227434926  0.227332295
## [6,] -0.001199444 -0.001192970 -0.001186511 -0.001180067 -0.001173639
## [7,]  0.005838678  0.005759411  0.005680422  0.005601708  0.005523270
##              [,36]        [,37]        [,38]        [,39]        [,40]
## [1,]  0.013829790  0.013846100  0.013862362  0.013878575  0.013894739
## [2,]  0.094736872  0.094616698  0.094496912  0.094377510  0.094258492
## [3,]  1.090537555  1.088750273  1.086969190  1.085194274  1.083425494
## [4,]  0.209283635  0.209574007  0.209863200  0.210151218  0.210438068
```

```
## [5,]  0.227229918  0.227127794  0.227025922  0.226924300  0.226822928
## [6,] -0.001167226 -0.001160828 -0.001154445 -0.001148078 -0.001141725
## [7,]  0.005445104  0.005367211  0.005289588  0.005212235  0.005135149
##            [,41]        [,42]        [,43]        [,44]        [,45]
## [1,]  0.013910856  0.013926924  0.013942945  0.013958919  0.013974845
## [2,]  0.094139856  0.094021598  0.093903719  0.093786214  0.093669083
## [3,]  1.081662816  1.079906210  1.078155644  1.076411086  1.074672506
## [4,]  0.210723757  0.211008290  0.211291674  0.211573915  0.211855018
## [5,]  0.226721803  0.226620926  0.226520293  0.226419905  0.226319761
## [6,] -0.001135387 -0.001129064 -0.001122756 -0.001116462 -0.001110183
## [7,]  0.005058330  0.004981776  0.004905486  0.004829459  0.004753692
##            [,46]        [,47]        [,48]        [,49]        [,50]
## [1,]  0.013990725  0.014006557  0.014022343  0.014038083  0.014053777
## [2,]  0.093552324  0.093435935  0.093319913  0.093204258  0.093088966
## [3,]  1.072939874  1.071213157  1.069492326  1.067777352  1.066068203
## [4,]  0.212134989  0.212413835  0.212691560  0.212968172  0.213243676
## [5,]  0.226219858  0.226120196  0.226020773  0.225921589  0.225822643
## [6,] -0.001103919 -0.001097668 -0.001091433 -0.001085211 -0.001079004
## [7,]  0.004678186  0.004602938  0.004527948  0.004453214  0.004378734
##            [,51]        [,52]        [,53]        [,54]        [,55]
## [1,]  0.014069426  0.014085028  0.014100586  0.014116098  0.014131565
## [2,]  0.092974038  0.092859469  0.092745259  0.092631406  0.092517909
## [3,]  1.064364850  1.062667263  1.060975414  1.059289273  1.057608810
## [4,]  0.213518077  0.213791381  0.214063594  0.214334721  0.214604768
## [5,]  0.225723932  0.225625457  0.225527215  0.225429206  0.225331429
## [6,] -0.001072811 -0.001066632 -0.001060468 -0.001054317 -0.001048180
## [7,]  0.004304508  0.004230535  0.004156812  0.004083340  0.004010116
##            [,56]        [,57]        [,58]        [,59]        [,60]
## [1,]  0.014146988  0.014162366  0.014177700  0.014192990  0.014208236
## [2,]  0.092404764  0.092291971  0.092179528  0.092067433  0.091955685
## [3,]  1.055933998  1.054264808  1.052601211  1.050943179  1.049290684
## [4,]  0.214873741  0.215141645  0.215408485  0.215674267  0.215938996
## [5,]  0.225233882  0.225136565  0.225039476  0.224942615  0.224845979
## [6,] -0.001042057 -0.001035947 -0.001029852 -0.001023770 -0.001017701
## [7,]  0.003937139  0.003864408  0.003791923  0.003719681  0.003647682
##            [,61]        [,62]         [,63]         [,64]         [,65]
## [1,]  0.014223439  0.014238598  0.0142537143  0.0142687876  0.0142838180
## [2,]  0.091844280  0.091733219  0.0916224991  0.0915121186  0.0914020759
## [3,]  1.047643699  1.046002195  1.0443661456  1.0427355230  1.0411103004
## [4,]  0.216202678  0.216465318  0.2167269217  0.2169874934  0.2172470387
## [5,]  0.224749569  0.224653383  0.2245574194  0.2244616781  0.2243661577
## [6,] -0.001011646 -0.001005604 -0.0009995762 -0.0009935614 -0.0009875597
## [7,]  0.003575924  0.003504407  0.0034331286  0.0033620882  0.0032912847
##             [,66]         [,67]         [,68]         [,69]         [,70]
## [1,]  0.0142988060  0.0143137516  0.0143286551  0.0143435168  0.0143583367
## [2,]  0.0912923694  0.0911829974  0.0910739583  0.0909652504  0.0908568722
## [3,]  1.0394904508  1.0378759474  1.0362667639  1.0346628737  1.0330642507
## [4,]  0.2175055629  0.2177630711  0.2180195684  0.2182750599  0.2185295506
## [5,]  0.2242708571  0.2241757754  0.2240809114  0.2239862643  0.2238918329
## [6,] -0.0009815712 -0.0009755959 -0.0009696336 -0.0009636842 -0.0009577479
## [7,]  0.0032207167  0.0031503833  0.0030802833  0.0030104156  0.0029407791
##             [,71]         [,72]         [,73]         [,74]         [,75]
## [1,]  0.0143731150  0.0143878521  0.0144025481  0.0144172031  0.0144318175
## [2,]  0.0907488219  0.0906410981  0.0905336991  0.0904266234  0.0903198694
```

```
## [3,]  1.0314708689  1.0298827024  1.0282997256  1.0267219129  1.0251492390
## [4,]  0.2187830455  0.2190355496  0.2192870679  0.2195376051  0.2197871663
## [5,]  0.2237976164  0.2237036136  0.2236098236  0.2235162454  0.2234228780
## [6,] -0.0009518243 -0.0009459136 -0.0009400157 -0.0009341304 -0.0009282578
## [7,]  0.0028713726  0.0028021951  0.0027332455  0.0026645227  0.0025960257
##               [,76]         [,77]         [,78]         [,79]        [,80]
## [1,]  0.0144463913  0.0144609247  0.0144754180  0.0144898714  0.014504285
## [2,]  0.0902134354  0.0901073201  0.0900015218  0.0898960390  0.089790870
## [3,]  1.0235816787  1.0220192071  1.0204617991  1.0189094303  1.017362076
## [4,]  0.2200357563  0.2202833798  0.2205300417  0.2207757466  0.221020499
## [5,]  0.2233297206  0.2232367720  0.2231440313  0.2230514976  0.222959170
## [6,] -0.0009223977 -0.0009165502 -0.0009107151 -0.0009048924 -0.000899082
## [7,]  0.0025277533  0.0024597045  0.0023918782  0.0023242735  0.002256889
##               [,81]        [,82]         [,83]         [,84]         [,85]
## [1,]  0.0145186590  0.014532994  0.0145472891  0.0145615455  0.0145757631
## [2,]  0.0896860139  0.089581469  0.0894772327  0.0893733049  0.0892696837
## [3,]  1.0158197118  1.014282314  1.0127498573  1.0112223190  1.0096996750
## [4,]  0.2212643047  0.221507167  0.2217490915  0.2219900821  0.2222301437
## [5,]  0.2228670475  0.222775129  0.2226834140  0.2225919012  0.2225005899
## [6,] -0.0008932839 -0.000887498 -0.0008817244 -0.0008759628 -0.0008702133
## [7,]  0.0021897243  0.002122778  0.0020560488  0.0019895360  0.0019232387
##               [,86]         [,87]         [,88]         [,89]         [,90]
## [1,]  0.0145899420  0.0146040824  0.0146181845  0.0146322485  0.0146462746
## [2,]  0.0891663675  0.0890633550  0.0889606447  0.0888582352  0.0887561250
## [3,]  1.0081819016  1.0066689754  1.0051608732  1.0036575717  1.0021590480
## [4,]  0.2224692808  0.2227074980  0.2229447996  0.2231811901  0.2234166740
## [5,]  0.2224094790  0.2223185676  0.2222278550  0.2221373402  0.2220470223
## [6,] -0.0008644757 -0.0008587502 -0.0008530365 -0.0008473346 -0.0008416446
## [7,]  0.0018571557  0.0017912860  0.0017256287  0.0016601828  0.0015949473
##               [,91]         [,92]         [,93]         [,94]         [,95]
## [1,]  0.0146602629  0.0146742136  0.0146881269  0.0147020029  0.0147158419
## [2,]  0.0886543127  0.0885527970  0.0884515764  0.0883506496  0.0882500151
## [3,]  1.0006652793  0.9991762427  0.9976919159  0.9962122763  0.9947373017
## [4,]  0.2236512557  0.2238849395  0.2241177298  0.2243496308  0.2245806469
## [5,]  0.2219569004  0.2218669736  0.2217772411  0.2216877021  0.2215983556
## [6,] -0.0008359662 -0.0008302996 -0.0008246446 -0.0008190011 -0.0008133692
## [7,]  0.0015299213  0.0014651037  0.0014004936  0.0013360901  0.0012718922
##               [,96]         [,97]         [,98]         [,99]        [,100]
## [1,]  0.0147296440  0.0147434093  0.0147571381  0.0147708305  0.0147844866
## [2,]  0.0881496716  0.0880496178  0.0879498522  0.0878503736  0.0877511806
## [3,]  0.9932669700  0.9918012591  0.9903401473  0.9888836127  0.9874316338
## [4,]  0.2248107823  0.2250400413  0.2252684280  0.2254959467  0.2257226014
## [5,]  0.2215092007  0.2214202368  0.2213314628  0.2212428780  0.2211544816
## [6,] -0.0008077487 -0.0008021396 -0.0007965419 -0.0007909556 -0.0007853804
## [7,]  0.0012078990  0.0011441095  0.0010805229  0.0010171381  0.0009539543
```

```
coef(glmnet_fit)
```

```
## 8 x 100 sparse Matrix of class "dgCMatrix"
##    [[ suppressing 100 column names 's0', 's1', 's2' ... ]]
##
## (Intercept)  4.762144e-17  4.763352e-17  4.764974e-17  4.766726e-17
## age          3.367377e-02  3.377901e-02  3.388779e-02  3.399735e-02
## lbph         4.800245e-02  4.830898e-02  4.861966e-02  4.893458e-02
```

```
## svi          2.069505e-01  2.078403e-01  2.087481e-01  2.096655e-01
## lcp          1.628635e-01  1.633983e-01  1.639341e-01  1.644739e-01
## gleason      6.500508e-02  6.499603e-02  6.497858e-02  6.495878e-02
## pgg45        8.347940e-02  8.349620e-02  8.351060e-02  8.352198e-02
## train       -1.646808e-02 -1.651265e-02 -1.655791e-02 -1.660328e-02
##
## (Intercept)  4.768614e-17  4.770642e-17  4.772811e-17  4.775127e-17
## age          3.410771e-02  3.421889e-02  3.433089e-02  3.444374e-02
## lbph         4.925380e-02  4.957742e-02  4.990553e-02  5.023823e-02
## svi          2.105926e-01  2.115296e-01  2.124767e-01  2.134341e-01
## lcp          1.650177e-01  1.655655e-01  1.661174e-01  1.666734e-01
## gleason      6.493653e-02  6.491176e-02  6.488441e-02  6.485441e-02
## pgg45        8.353025e-02  8.353532e-02  8.353710e-02  8.353547e-02
## train       -1.664874e-02 -1.669430e-02 -1.673994e-02 -1.678567e-02
##
## (Intercept)  4.777593e-17  4.780212e-17  4.782988e-17  4.785924e-17
## age          3.455743e-02  3.467197e-02  3.478739e-02  3.490369e-02
## lbph         5.057562e-02  5.091779e-02  5.126485e-02  5.161691e-02
## svi          2.144021e-01  2.153807e-01  2.163702e-01  2.173708e-01
## lcp          1.672336e-01  1.677981e-01  1.683668e-01  1.689398e-01
## gleason      6.482171e-02  6.478624e-02  6.474792e-02  6.470669e-02
## pgg45        8.353034e-02  8.352158e-02  8.350910e-02  8.349278e-02
## train       -1.683148e-02 -1.687736e-02 -1.692329e-02 -1.696929e-02
##
## (Intercept)  4.789026e-17  4.792297e-17  4.795742e-17  4.799364e-17
## age          3.502088e-02  3.513896e-02  3.525796e-02  3.537789e-02
## lbph         5.197408e-02  5.233647e-02  5.270419e-02  5.307738e-02
## svi          2.183828e-01  2.194063e-01  2.204416e-01  2.214889e-01
## lcp          1.695173e-01  1.700992e-01  1.706856e-01  1.712766e-01
## gleason      6.466247e-02  6.461519e-02  6.456478e-02  6.451116e-02
## pgg45        8.347248e-02  8.344810e-02  8.341949e-02  8.338653e-02
## train       -1.701533e-02 -1.706141e-02 -1.710752e-02 -1.715365e-02
##
## (Intercept)  4.803168e-17  4.807158e-17  4.811340e-17  4.815718e-17
## age          3.549875e-02  3.562055e-02  3.574332e-02  3.586705e-02
## lbph         5.345614e-02  5.384062e-02  5.423094e-02  5.462724e-02
## svi          2.225485e-01  2.236205e-01  2.247053e-01  2.258032e-01
## lcp          1.718722e-01  1.724724e-01  1.730775e-01  1.736873e-01
## gleason      6.445424e-02  6.439396e-02  6.433021e-02  6.426292e-02
## pgg45        8.334907e-02  8.330698e-02  8.326010e-02  8.320828e-02
## train       -1.719978e-02 -1.724592e-02 -1.729205e-02 -1.733815e-02
##
## (Intercept)  4.820297e-17  4.825081e-17  4.830077e-17  4.835290e-17
## age          3.599177e-02  3.611749e-02  3.624421e-02  3.637195e-02
## lbph         5.502966e-02  5.543834e-02  5.585344e-02  5.627510e-02
## svi          2.269143e-01  2.280391e-01  2.291777e-01  2.303305e-01
## lcp          1.743020e-01  1.749216e-01  1.755463e-01  1.761760e-01
## gleason      6.419200e-02  6.411736e-02  6.403891e-02  6.395655e-02
## pgg45        8.315137e-02  8.308919e-02  8.302157e-02  8.294834e-02
## train       -1.738421e-02 -1.743023e-02 -1.747618e-02 -1.752205e-02
##
## (Intercept)  4.840724e-17  4.846387e-17  4.852282e-17  4.858417e-17
## age          3.650073e-02  3.663055e-02  3.676143e-02  3.689339e-02
## lbph         5.670349e-02  5.713877e-02  5.758111e-02  5.803068e-02
```

```
## svi          2.314977e-01  2.326797e-01  2.338769e-01  2.350895e-01
## lcp          1.768108e-01  1.774509e-01  1.780962e-01  1.787469e-01
## gleason      6.387018e-02  6.377971e-02  6.368503e-02  6.358605e-02
## pgg45        8.286931e-02  8.278429e-02  8.269308e-02  8.259548e-02
## train       -1.756782e-02 -1.761349e-02 -1.765902e-02 -1.770440e-02
##
## (Intercept)  4.864797e-17  4.871429e-17  4.878318e-17  4.885473e-17
## age          3.702642e-02  3.716056e-02  3.729581e-02  3.743219e-02
## lbph         5.848768e-02  5.895228e-02  5.942467e-02  5.990507e-02
## svi          2.363179e-01  2.375625e-01  2.388237e-01  2.401017e-01
## lcp          1.794030e-01  1.800646e-01  1.807318e-01  1.814046e-01
## gleason      6.348264e-02  6.337470e-02  6.326212e-02  6.314478e-02
## pgg45        8.249127e-02  8.238023e-02  8.226212e-02  8.213672e-02
## train       -1.774962e-02 -1.779465e-02 -1.783947e-02 -1.788405e-02
##
## (Intercept)  4.892899e-17  4.900603e-17  4.908592e-17  4.916875e-17
## age          3.756971e-02  3.770838e-02  3.784822e-02  3.798924e-02
## lbph         6.039367e-02  6.089069e-02  6.139636e-02  6.191090e-02
## svi          2.413971e-01  2.427102e-01  2.440415e-01  2.453913e-01
## lcp          1.820831e-01  1.827675e-01  1.834577e-01  1.841539e-01
## gleason      6.302257e-02  6.289536e-02  6.276303e-02  6.262545e-02
## pgg45        8.200376e-02  8.186299e-02  8.171414e-02  8.155693e-02
## train       -1.792837e-02 -1.797241e-02 -1.801613e-02 -1.805951e-02
##
## (Intercept)  4.925458e-17  4.934348e-17  4.943555e-17  4.953086e-17
## age          3.813145e-02  3.827488e-02  3.841953e-02  3.856541e-02
## lbph         6.243455e-02  6.296756e-02  6.351019e-02  6.406270e-02
## svi          2.467603e-01  2.481487e-01  2.495572e-01  2.509862e-01
## lcp          1.848561e-01  1.855645e-01  1.862791e-01  1.870000e-01
## gleason      6.248250e-02  6.233403e-02  6.217991e-02  6.202001e-02
## pgg45        8.139106e-02  8.121623e-02  8.103211e-02  8.083839e-02
## train       -1.810251e-02 -1.814511e-02 -1.818726e-02 -1.822893e-02
##
## (Intercept)  4.962950e-17  4.973154e-17  4.983708e-17  4.994622e-17
## age          3.871255e-02  3.886096e-02  3.901064e-02  3.916161e-02
## lbph         6.462537e-02  6.519848e-02  6.578233e-02  6.637723e-02
## svi          2.524362e-01  2.539079e-01  2.554017e-01  2.569182e-01
## lcp          1.877273e-01  1.884610e-01  1.892013e-01  1.899483e-01
## gleason      6.185417e-02  6.168226e-02  6.150412e-02  6.131960e-02
## pgg45        8.063471e-02  8.042072e-02  8.019603e-02  7.996025e-02
## train       -1.827009e-02 -1.831068e-02 -1.835066e-02 -1.838999e-02
##
## (Intercept)  5.005903e-17  5.017562e-17  5.029609e-17  5.042052e-17
## age          3.931389e-02  3.946749e-02  3.962242e-02  3.977870e-02
## lbph         6.698350e-02  6.760147e-02  6.823149e-02  6.887392e-02
## svi          2.584581e-01  2.600220e-01  2.616105e-01  2.632244e-01
## lcp          1.907021e-01  1.914626e-01  1.922302e-01  1.930047e-01
## gleason      6.112854e-02  6.093080e-02  6.072619e-02  6.051457e-02
## pgg45        7.971297e-02  7.945377e-02  7.918220e-02  7.889778e-02
## train       -1.842862e-02 -1.846649e-02 -1.850355e-02 -1.853973e-02
##
## (Intercept)  5.054904e-17  5.068173e-17  5.081872e-17  5.096010e-17
## age          3.993633e-02  4.009533e-02  4.025570e-02  4.041747e-02
## lbph         6.952914e-02  7.019753e-02  7.087950e-02  7.157547e-02
```

```
## svi          2.648643e-01  2.665310e-01  2.682253e-01  2.699480e-01
## lcp          1.937864e-01  1.945753e-01  1.953715e-01  1.961751e-01
## gleason      6.029576e-02  6.006958e-02  5.983587e-02  5.959445e-02
## pgg45        7.860003e-02  7.828843e-02  7.796245e-02  7.762151e-02
## train       -1.857497e-02 -1.860921e-02 -1.864237e-02 -1.867437e-02
##
## (Intercept)  5.110599e-17  5.125651e-17  5.141177e-17  5.157190e-17
## age          4.058063e-02  4.074520e-02  4.091118e-02  4.107858e-02
## lbph         7.228589e-02  7.301122e-02  7.375193e-02  7.450853e-02
## svi          2.716998e-01  2.734817e-01  2.752946e-01  2.771395e-01
## lcp          1.969862e-01  1.978049e-01  1.986314e-01  1.994656e-01
## gleason      5.934513e-02  5.908774e-02  5.882209e-02  5.854800e-02
## pgg45        7.726503e-02  7.689239e-02  7.650293e-02  7.609598e-02
## train       -1.870513e-02 -1.873456e-02 -1.876258e-02 -1.878907e-02
##
## (Intercept)  5.173700e-17  5.190722e-17  5.208267e-17  5.224873e-17
## age          4.124741e-02  4.141767e-02  4.158936e-02  4.175790e-02
## lbph         7.528153e-02  7.607150e-02  7.687899e-02  7.769621e-02
## svi          2.790172e-01  2.809290e-01  2.828759e-01  2.848125e-01
## lcp          2.003077e-01  2.011578e-01  2.020160e-01  2.029000e-01
## gleason      5.826527e-02  5.797372e-02  5.767315e-02  5.738901e-02
## pgg45        7.567081e-02  7.522666e-02  7.476275e-02  7.427512e-02
## train       -1.881393e-02 -1.883706e-02 -1.885831e-02 -1.887462e-02
##
## (Intercept)  5.243428e-17  5.262522e-17  5.282186e-17  5.302433e-17
## age          4.193220e-02  4.210782e-02  4.228484e-02  4.246326e-02
## lbph         7.854066e-02  7.940454e-02  8.028855e-02  8.119340e-02
## svi          2.868315e-01  2.888886e-01  2.909857e-01  2.931240e-01
## lcp          2.037759e-01  2.046606e-01  2.055537e-01  2.064555e-01
## gleason      5.707122e-02  5.674429e-02  5.640771e-02  5.606128e-02
## pgg45        7.376836e-02  7.323894e-02  7.268601e-02  7.210853e-02
## train       -1.889145e-02 -1.890591e-02 -1.891789e-02 -1.892721e-02
##
## (Intercept)  5.323277e-17  5.344730e-17  5.366807e-17  5.389521e-17
## age          4.264304e-02  4.282419e-02  4.300667e-02  4.319046e-02
## lbph         8.211985e-02  8.306871e-02  8.404080e-02  8.503702e-02
## svi          2.953050e-01  2.975303e-01  2.998013e-01  3.021200e-01
## lcp          2.073660e-01  2.082852e-01  2.092133e-01  2.101503e-01
## gleason      5.570485e-02  5.533823e-02  5.496127e-02  5.457382e-02
## pgg45        7.150538e-02  7.087540e-02  7.021735e-02  6.952990e-02
## train       -1.893368e-02 -1.893711e-02 -1.893727e-02 -1.893393e-02
##
## (Intercept)  5.412884e-17  5.436909e-17  5.461610e-17  5.486998e-17
## age          4.337552e-02  4.356183e-02  4.374933e-02  4.393796e-02
## lbph         8.605827e-02  8.710555e-02  8.817986e-02  8.928230e-02
## svi          3.044878e-01  3.069069e-01  3.093791e-01  3.119065e-01
## lcp          2.110963e-01  2.120512e-01  2.130151e-01  2.139880e-01
## gleason      5.417574e-02  5.376688e-02  5.334715e-02  5.291643e-02
## pgg45        6.881166e-02  6.806116e-02  6.727682e-02  6.645697e-02
## train       -1.892683e-02 -1.891570e-02 -1.890024e-02 -1.888015e-02
##
## (Intercept)  5.513085e-17  5.539884e-17  5.567403e-17  5.595652e-17
## age          4.412768e-02  4.431841e-02  4.451007e-02  4.470257e-02
## lbph         9.041401e-02  9.157620e-02  9.277013e-02  9.399716e-02
```

```
## svi          3.144912e-01  3.171358e-01  3.198425e-01  3.226141e-01
## lcp          2.149700e-01  2.159610e-01  2.169609e-01  2.179698e-01
## gleason      5.247466e-02  5.202178e-02  5.155777e-02  5.108265e-02
## pgg45        6.559982e-02  6.470347e-02  6.376590e-02  6.278494e-02
## train       -1.885506e-02 -1.882461e-02 -1.878838e-02 -1.874595e-02
##
## (Intercept)  5.624641e-17  5.654375e-17  5.684861e-17  5.716101e-17
## age          4.489580e-02  4.508964e-02  4.528396e-02  4.547858e-02
## lbph         9.525872e-02  9.655632e-02  9.789157e-02  9.926618e-02
## svi          3.254534e-01  3.283631e-01  3.313467e-01  3.344073e-01
## lcp          2.189874e-01  2.200137e-01  2.210485e-01  2.220916e-01
## gleason      5.059645e-02  5.009928e-02  4.959130e-02  4.907270e-02
## pgg45        6.175826e-02  6.068340e-02  5.955767e-02  5.837824e-02
## train       -1.869683e-02 -1.864049e-02 -1.857637e-02 -1.850386e-02
##
## (Intercept)  5.748096e-17  5.780846e-17  5.814345e-17  5.848583e-17
## age          4.567335e-02  4.586805e-02  4.606245e-02  4.625630e-02
## lbph         1.006820e-01  1.021409e-01  1.036449e-01  1.051963e-01
## svi          3.375486e-01  3.407744e-01  3.440888e-01  3.474962e-01
## lcp          2.231427e-01  2.242016e-01  2.252678e-01  2.263410e-01
## gleason      4.854377e-02  4.800489e-02  4.745650e-02  4.689920e-02
## pgg45        5.714201e-02  5.584568e-02  5.448568e-02  5.305815e-02
## train       -1.842226e-02 -1.833084e-02 -1.822879e-02 -1.811522e-02
##
## (Intercept)  5.883548e-17  5.919221e-17  5.955576e-17  5.987819e-17
## age          4.644929e-02  4.664108e-02  4.683130e-02  4.701549e-02
## lbph         1.067975e-01  1.084509e-01  1.101592e-01  1.119111e-01
## svi          3.510013e-01  3.546092e-01  3.583253e-01  3.620726e-01
## lcp          2.274205e-01  2.285058e-01  2.295960e-01  2.307427e-01
## gleason      4.633367e-02  4.576077e-02  4.518154e-02  4.467973e-02
## pgg45        5.155890e-02  4.998341e-02  4.832676e-02  4.654234e-02
## train       -1.798913e-02 -1.784945e-02 -1.769498e-02 -1.750823e-02
##
## (Intercept)  6.025075e-17  6.062804e-17  6.100987e-17  6.139540e-17
## age          4.720065e-02  4.738266e-02  4.756087e-02  4.773453e-02
## lbph         1.137384e-01  1.156303e-01  1.175902e-01  1.196221e-01
## svi          3.660203e-01  3.700943e-01  3.743032e-01  3.786551e-01
## lcp          2.318443e-01  2.329497e-01  2.340562e-01  2.351621e-01
## gleason      4.409827e-02  4.351630e-02  4.293524e-02  4.235762e-02
## pgg45        4.470155e-02  4.276044e-02  4.071273e-02  3.855062e-02
## train       -1.731827e-02 -1.710861e-02 -1.687766e-02 -1.662336e-02
##
## (Intercept)  6.178366e-17  6.217344e-17  6.256330e-17  6.295151e-17
## age          4.790278e-02  4.806463e-02  4.821895e-02  4.836445e-02
## lbph         1.217304e-01  1.239195e-01  1.261947e-01  1.285613e-01
## svi          3.831590e-01  3.878248e-01  3.926630e-01  3.976856e-01
## lcp          2.362657e-01  2.373646e-01  2.384563e-01  2.395380e-01
## gleason      4.178639e-02  4.122505e-02  4.067773e-02  4.014927e-02
## pgg45        3.626551e-02  3.384794e-02  3.128738e-02  2.857218e-02
## train       -1.634339e-02 -1.603519e-02 -1.569585e-02 -1.532207e-02
##
## (Intercept)  6.333599e-17  6.365030e-17  6.401263e-17  6.436087e-17
## age          4.849962e-02  4.862369e-02  4.873210e-02  4.882416e-02
## lbph         1.310255e-01  1.335729e-01  1.362529e-01  1.390526e-01
```

```
## svi          4.029057e-01  4.082508e-01  4.139069e-01  4.198076e-01
## lcp          2.406059e-01  2.417112e-01  2.427445e-01  2.437519e-01
## gleason      3.964539e-02  3.928333e-02  3.886253e-02  3.849268e-02
## pgg45        2.568932e-02  2.255998e-02  1.928554e-02  1.579142e-02
## train       -1.491015e-02 -1.443215e-02 -1.392703e-02 -1.336838e-02
```

```
c(total_time1,total_time2)
```

```
## Time differences in secs
## [1] 0.02018571 0.01624608
```

We could find that ridge regression by data augmentation is easy to understand and precise, ridge regression by data augmentation would not consider the β_0, but native implementation of ridge regression does. svi is the most significant factor for ridge regression by data augmentation, other factors are not significant, ative implementation of ridge regression would take every factor almost equally significant.

also, the running time of native implementation of ridge regression(0.04307628) is longer than ridge regression by data augmentation(0.02826595).

ex3.6

In ridge regression, we want to minimize $||\vec{y} - X\vec{\beta}||_2^2 + \lambda||\vec{\beta}||_2^2$

The posterior distribution of $\vec{\beta}$ given $\vec{y}$ is proportional to the product of the likelihood and the prior:

$p(\vec{\beta}|\vec{y}) \propto p(\vec{y}|\vec{\beta})p(\vec{\beta})$

$p(\vec{y}|\vec{\beta}) \propto exp(-\frac{1}{2\sigma^2}||\vec{y} - X\vec{\beta}||_2^2)exp(-\frac{1}{2\tau}||\vec{\beta}||_2^2)$

$-log(p(\vec{y}|\vec{\beta})) = \frac{1}{2\sigma^2}||\vec{y} - X\vec{\beta}||_2^2 + \frac{1}{2\tau}||\vec{\beta}||_2^2$

The ridge regression objective function is the negative log posterior. Therefore, minimizing the ridge regression objective function is equivalent to maximizing the posterior distribution, which means we want to find the mode of the posterior distribution. By the properties of the Gaussian distribution, the mode and mean of the posterior distribution are the same, so the ridge regression estimate is also the mean of the posterior distribution.

From the above, we could see that $\hat{\vec{\beta}} = (X'X + \frac{\sigma^2}{\tau}I)^{-1}X'\vec{y}$, therefore, $\lambda = \frac{\sigma^2}{\tau}$.

ex3.19

As we know, $\hat{\beta}^{ridge} = (X'X + \lambda I)^{-1}X'\vec{y}$, it could also be written as $\frac{X'\vec{y}}{X'X+\lambda I}$; therefore, we could find that as λ decrease, $\hat{\beta}^{ridge}$ will increase, which means $\lambda \to 0$, $\hat{\beta}^{ridge}$ will increase.

For lasso, $\hat{\beta}^{lasso} = argmin_{\beta}\frac{1}{2n}||\vec{y} - X\vec{\beta}||_2^2 + \alpha||\beta||_1$, where α is the tuning parameter, the size of the updates to the coefficients depends on the value of α, with larger values of α leading to larger updates, we could see that when α decrease, $\hat{\beta}^{lasso}$ will also decrease. So, the same property does not hold for lasso.

ex3.28

suppose we have a copy of X named X^*, the new predictor would be $X^N = [X, X^*]$.

then the new coefficient would be $\beta^N = (\beta', \beta^{*\prime})$, $X^N\beta^N = X\beta + X^*\beta^* = X(\beta + \beta^*)$

$L = ||\vec{y} - X^N\beta^N||^2 + \lambda(\sum_{j=1}^p |\beta_j| + \sum_{j=1}^p |\beta_j^*|) = ||\vec{y} - X(\beta + \beta^*)||^2 + \lambda(\sum_{j=1}^p |\beta_j + \beta_j^*|) - \lambda(\sum_{j=1}^p |\beta_j + \beta_j^*|) + \lambda(\sum_{j=1}^p |\beta_j| + \sum_{j=1}^p |\beta_j^*|)$

the first two term is the same problem for the lossa, and the last two term would be non negative since |a+b|≤ |a|+|b|

By 3.52, we could know that the solution of the first two term would be $\beta + \beta^* = \hat{\beta}$

So, $\beta_j + \beta_j^* = \hat{\beta}_j = \alpha$

and in the same time, we want the last two term to be zero, so β and β^* should be both positive or negative.

ex3.30

suppose we have a new $\bar{X} = (X, \gamma I)'$ and $\bar{y} = (y, 0)'$ where I is identity matrix and $\gamma$is a constant number

$||\bar{y} - \bar{X}\beta||_2^2 = ||(y - X\beta, -\gamma\beta)||_2^2 = ||y - X\beta||_2^2 + \gamma^2||\beta||_2^2$

$\hat{\beta} = argmin_\beta(||\bar{y} - \bar{X}\beta||_2^2 + \bar{\lambda}||\beta||_1) = argmin_\beta(||y - X\beta||_2^2 + \gamma^2||\beta||_2^2 + \bar{\lambda}||\beta||_1)$

from 3.91, we know that we want $min_\beta||y - X\beta||^2 + \lambda(\alpha||\beta||_2^2 + (1 - \alpha)||\beta||_1)$

we could see that$\alpha\lambda = \gamma^2, (1 - \alpha)\lambda = \bar{\lambda}$

so $\gamma = \sqrt{\alpha\lambda}, \bar{\lambda} = (1 - \alpha)\lambda$