Homework #6 **20180047 김경수**

0. 이론적 배경

(0) Feature selection

앞으로 설명할 전진 단계적 선택이나 shrinkage method들은 일반적인 최소제곱적합을 대체하는 보다 효율적인 방법이다. 이러한 적합절차를 사용하려는 이유는 bias와 variance의 관점에서 볼 때 더 나은 예측 정확도와 모델 해석력을 제공할 수 있기 때문이다. 크게 subset selection, shrinkage, dimension reduction 으로 나눌 수 있는데, 본 레포트에서는 forward stepwise feature selection, shirnkage(Lasso, Ridge Regression)을 사용하였다.

(1) Forward stepwise feature selection

알고리즘은 다음과 같다.

1. M0를 설명변수를 하나도 포함하지 않는 영모델이라고 하자.
2. k=0, 1, 2, ,,,, p-1에 대하여 (총 설명변수 개수 : p)
   (a) Mk에 하나의 설명변수를 추가한 p-k개의 모델 중 최고의 성능을 가진 모델을 M(k+1)이라 하자. (가장 작은 RSS나 가장 큰 R^2을 가지는 모델)
3. 교차검증된 MSE, Cp, AIC, BIC, adjusted R^2 등을 이용하여 M0,,,,Mp 중 최고의 모델을 선정한다.

본 레포트에서는 adjusted R^2을 이용하여 최고의 모델을 선정하였다. 일반적인 R^2은 설명변수의 수가 증가하면 R^2값도 증가하지만 adjusted r^2 의 경우 1-RSS/n-p-1)/(TSS/n-1)로 표현되기에, adjusted r^2의 경우 noise 변수들을 포함할 시, 그 값이 감소한다. 따라서 adjusted r^2이 최대인 모델을 선택하는 것으로 model selection을 진행하였다.

(2) Ridge Regression & Lasso

Ridge와 Lasso는 그 방식이 최소제곱적합과 매우 비슷하다.

Ridge의 경우 RSS와 shrinking penalty term(lambda * sum of coefficient's square)의 합을 minimize하는 model을 select한다. Lambda가 증가할수록 coefficient는 0으로 shrink하며, cross validation을 진행하여 best lambda를 선택한다.

Lasso의 경우 RSS+labmda * sum of absolute(coefficient)의 값을 minimized하는 모델을 선택한다. Ridge와 마찬가지로 lambda가 증가할수록 coefficient는 0으로 shrink하며, cross validation을 진행하여 best lambda를 선택한다.

RIdge의 경우 계수가 아예 0으로 수축하지는 않지만, Lasso의 경우 coefficient가 0으로 아예 수축가능하다. RIdge와 Lasso의 경우 어떤 방법을 선택할 지는, 각 상황에 따라 다르며, noise 변수가 많을 경우 Lasso, 그렇지 않고 각 설명변수가 균등한 영향을 반응변수에 미칠 경우에는 Ridge를 선택하는 것이 유리할 것이다.

***데이터 전처리에 관한 코드이다.**

```
clin <- readRDS("clinical.rds")
gex <- readRDS("expression.rds")
```

#clinical과 expression.rds 데이터들을 위와 같이 implement함.

```
clin <- clin[clin$survival_time!= 0, ]
idx <- intersect(colnames(gex), clin$sample)
clin_p <- clin[clin$sample_id %in% idx, ]
surv_time <- clin_p$survival_tim
gex_p <- gex[, colnames(gex) %in% idx]
gex_p <- na.omit(gex_p)
temp <- normalize.quantiles(gex_p)
colnames(temp) <- colnames(gex_p)
rownames(temp) <- rownames(gex_p)
gex_p <- temp
```

#Clinical, expression 데이터들을 filtering함. Survival time이 0인 data와 subtype이 NA인 data를 제거하고, Quantile Normalization을 진행한다.

```
cor.p <- cor.coef <- cor.idx <- c()
for(i in 1:dim(gex_p)[1]){
  cor.p <- c(cor.p, cor.test(clin_p$survival_time, gex_p[i, ],
method="pearson")$p.value)
  cor.coef <- c(cor.coef, cor.test(clin_p$survival_time, gex_p[i, ],
method="pearson")$estimate)
}
cor.idx <- order(cor.p, decreasing=FALSE)[1:100]
gex_p.cor <- t(gex_p[cor.idx, ])
data.cor <- data.frame(surv_time, gex_p.cor)
```

#계산량을 감소시키기 위해, survival time과의 correlation 계수가 높은 top 100 gene에 대한 selection을 진행해 data.cor 이라는 data frame에 이를 저장한다.

Homework #6

1. Find optimal number of predictors which minimize adjusted r square using forward stepwise feature selection. (10 Fold CV)
2. Compare Ridge Regression & Lasso model. Which one is better. Discuss its reason in terms of lambda. (Regularization Intensity)

1. Procedure & R code

(1) Forward stepwise selection을 진행한다. 이론적 배경에 서술한 M0,,,,Mp 모델을 선정하는 과정이다.

R code :

```
regfit <- regsubsets(surv_time ~ ., data = data.cor, method="forward", nvmax = 100)
reg.sum <- summary(regfit)
```

(2) Adjusted R^2 값과 number of predictor에 대한 plot을 그리고, maximum adjusted R^2 값을 가지는 최적의 number of predictor를 찾는다.

R code :

```
par(mfrow=c(1,2))
plot(reg.sum$adjr2, type = 'l', col = "red",
     xlab = "Number of Predictors", ylab = "Adjusted RSq")
points(reg.sum$adjr2, col = "red", pch = 16)
which(reg.sum$adjr2==max(reg.sum$adjr2))
```

(3) Ridge regression을 진행한다. Log lambda에 대한 MSE의 plot을 그리고, 최적의 lambda 값을 확인한다. 또한 이 lambda 값에 대한 final model을 확인한다.(계수를 확인)

R code :

```
x <- gex_p.cor
y <- surv_time
grid <- 10^seq(10, -2, length = 100)
ridge <- glmnet(x, y, alpha = 0, lambda = grid, standardize = TRUE)
set.seed(1)
train <- sample(1:nrow(x), nrow(x)/2)
test <- (-train)
y.test <- y[test]
ridge.mod <- glmnet(x[train,], y[train], alpha=0, lambda = grid)
cv.ridge <- cv.glmnet(x[train, ], y[train], alpha=0, nfolds=10)
plot(cv.ridge)
bestlambda <- cv.ridge$lambda.1se
bestlambda
out <- glmnet(x, y, alpha = 0, lambda = grid)
predict(out, type = "coefficients", s = bestlambda)
```

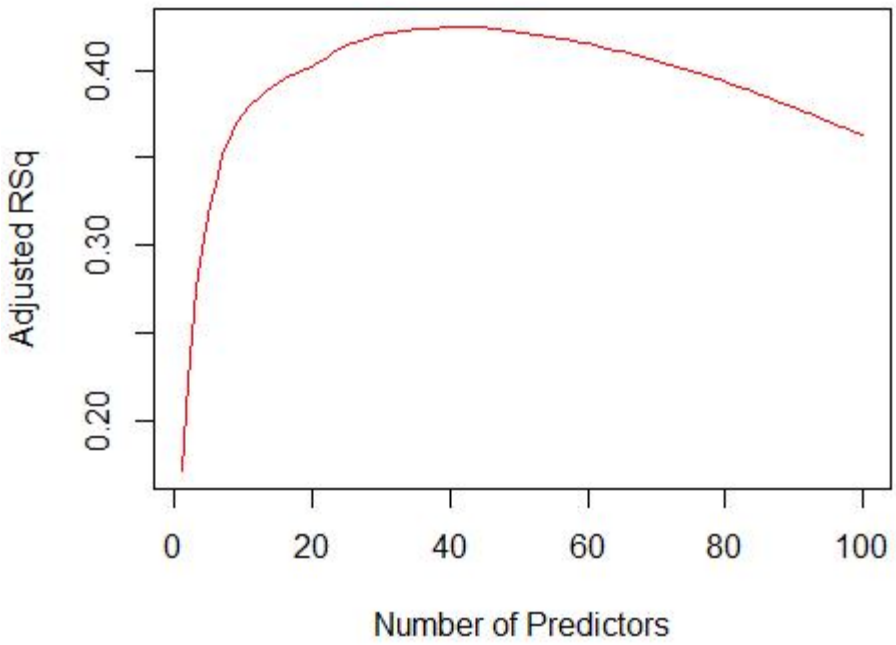
(4) Lasso method를 진행한다. Log lambda에 대한 MSE의 plot을 그리고, 최적의 lambda 값을 확인한다. 또한 이 lambda 값에 대한 final model을 확인한다.(계수를 확인)

R code :

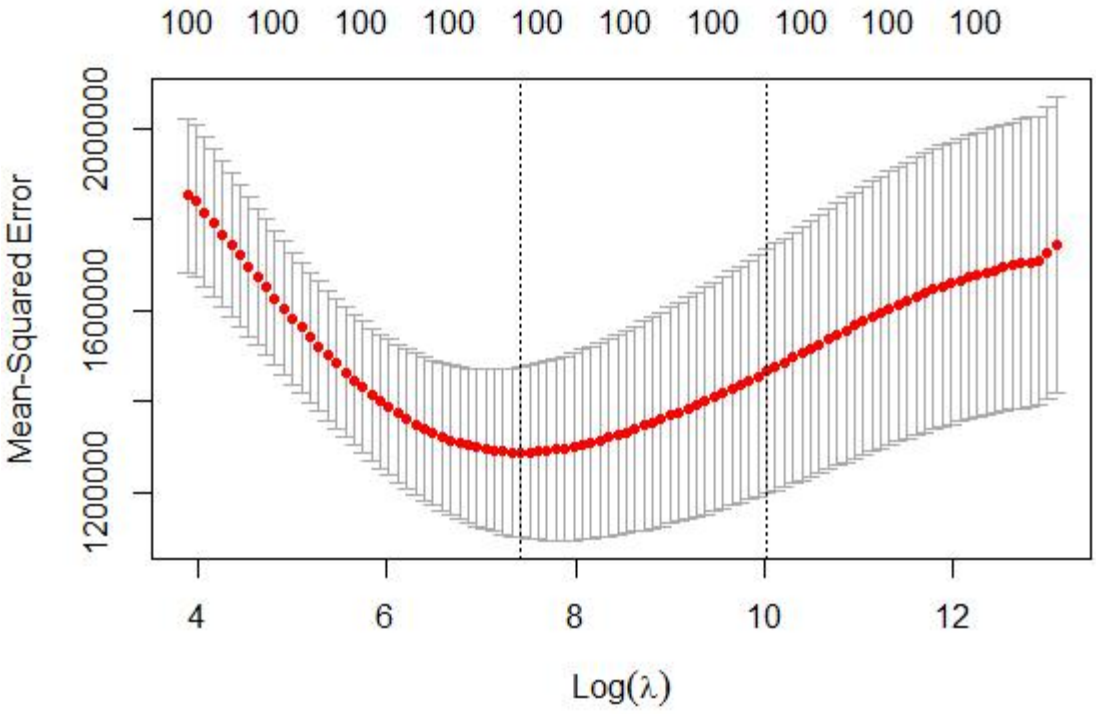
```
lasso <- glmnet(x, y, alpha = 1, lambda = grid, standardize = TRUE)
lasso.mod <- glmnet(x[train,], y[train], alpha=1, lambda = grid)
cv.lasso <- cv.glmnet(x[train, ], y[train], alpha=1, nfolds=10)
plot(cv.lasso)
bestlambda <- cv.lasso$lambda.1se
bestlambda
out <- glmnet(x, y, alpha = 1, lambda = grid)
predict(out, type = "coefficients", s = bestlambda)
```

2. Results

(1) Forward stepwise selection에 대한, number of predictors와 Adjusted R^2의 plot은 다음과 같으며, maximum adjusted R^2 값을 가지는 최적의 number of predictors 는 40이다.



(2) Ridge Regression을 진행한 결과, 최적 모델의 lambda 값은 22558.69로 측정되었다. Log lambda에 대한 MSE의 plot과 최적 모델의 결과(계수)는 다음과 같다.



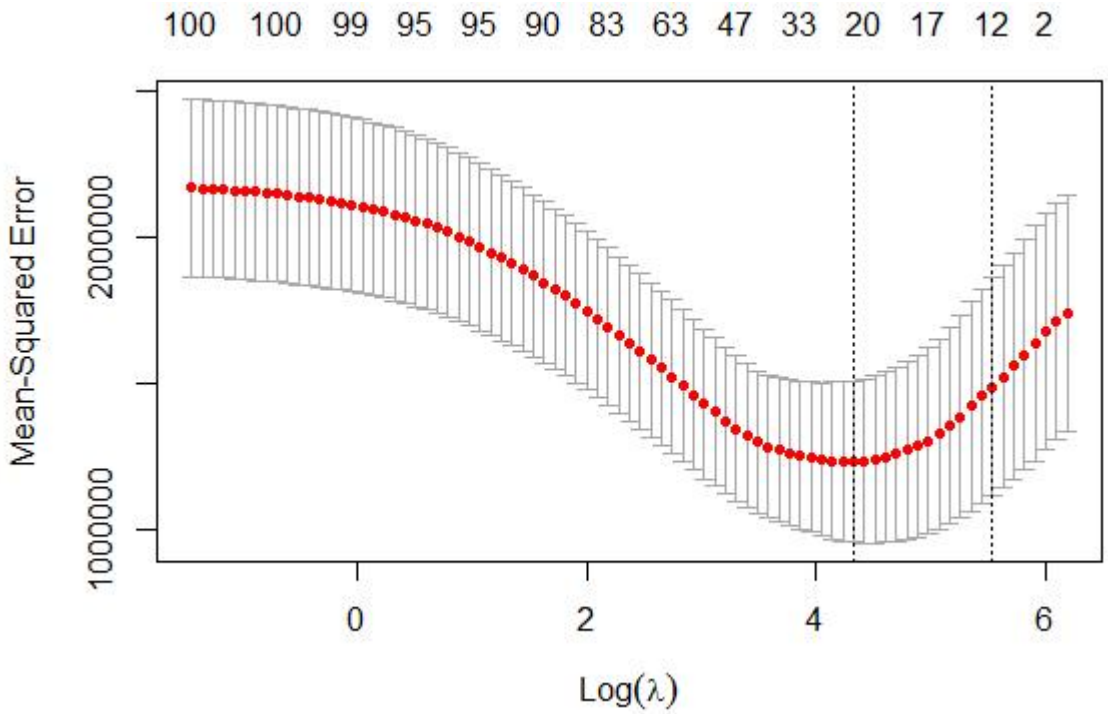
101 x 1 sparse Matrix of class "dgCMatrix"

	1
(Intercept)	2716.420066
FABP1	22.586054
TDGF1	23.645750
OR1D2	18.855598
G6PC	-23.768084
IAPP	-19.138797
CLK1	12.694221
KIRREL	-22.455244
COX7A1	-20.553469
ANKS4B	17.129914
MEFV	-41.248937
LIMK1	-7.568302
PAPOLA	-23.997423
TGFBRAP1	-21.751630
NKAPL	7.305226
TLR2	-18.198342
DNTT	-26.611887
GCKR	22.024082
GAST	-16.983176
CD14	-21.266797
GYPB	19.723523
SERPIND1	-36.099003
KLRC4	11.667449
SHBG	26.587995
SET	-12.555037
PRKCI	-9.665258
AMHR2	16.817226
RSPO2	-12.449559
TNFSF9	17.789670
LMNB1	-3.496329
FAM131C	-13.473306
LHFPL2	-24.488517
SLC39A5	25.707153
MFHAS1	-10.336259
LOC440348	16.647830
SERPINF2	15.236522
KIAA1394	26.412173
THOC5	-25.264137
TYROBP	30.050027
REST	-10.195172
VIL1	19.670067
CSAD	9.514834
ALDH18A1	-7.804533
FLJ25404	-23.940013
JMJD2A	-7.477360
HGFAC	12.588001
STX6	-7.876869
KIAA1505	10.841235
GIPR	-20.637684

C10orf96	-11.593427
PRAMEF8	24.729557
MAL	14.636986
CGB2	22.826950
APP	-9.624286
LGALS4	15.737021
C1orf142	-17.252588
ASGR1	12.235145
MST1	10.172292
LCAT	12.750894
AHSG	19.406958
F2	17.639594
CKAP2L	-3.552137
TTR	12.647383
SSR1	-12.086485
SLC26A10	8.888528
IL17RA	-20.480180
C14orf162	-16.202736
CKAP4	-8.822063
CTCFL	-9.141579
STAB1	-18.334181
HBG1	9.482875
CALCOCO1	10.902370
HIRA	9.946649
IFRG15	-12.341111
FAM111B	-2.488020
NHN1	-8.969957
TMPRSS11F	-19.567273
OR5J2	-14.551320
SIGLEC10	-11.173545
ATF6	-17.555168
NLGN2	24.713490
DPPA4	-23.651933
CCDC66	11.266614
LIN28B	-9.435420
SPATA13	-6.197313
RPL11	9.378541
TRIM59	-7.846461
LRRIQ2	-8.365248
ITIH3	13.408647
DDX6	-7.142200
LOC440248	6.240187
PMS2CL	33.410240
P2RY4	-10.559539
FCGR3A	-10.072241
RAG2	-12.000948
RSPO3	4.852918
ZNF509	16.803478
CAV3	12.130014
ANXA6	9.554169
SLC15A3	-10.128133

LARP7 15.389818

(2) Lasso method를 진행한 결과, 최적 모델의 lambda 값은 253.4071로 측정되었다. Log lambda에 대한 MSE의 plot과 최적 모델의 결과(계수)는 다음과 같다.



101 x 1 sparse Matrix of class "dgCMatrix"

	1
(Intercept)	3733.377373
FABP1	339.118040
TDGF1	103.134196
OR1D2	.
G6PC	.
IAPP	.
CLK1	18.543823
KIRREL	-46.563463
COX7A1	.
ANKS4B	.
MEFV	-82.622077
LIMK1	.
PAPOLA	.
TGFBRAP1	.
NKAPL	.
TLR2	-31.538888
DNTT	.
GCKR	.
GAST	.
CD14	.
GYPB	14.597392
SERPIND1	-151.025234
KLRC4	.

SHBG	.
SET	.
PRKCI	.
AMHR2	.
RSPO2	.
TNFSF9	.
LMNB1	.
FAM131C	.
LHFPL2	-25.565330
SLC39A5	.
MFHAS1	.
LOC440348	.
SERPINF2	.
KIAA1394	4.073979
THOC5	-1.902335
TYROBP	.
REST	.
VIL1	.
CSAD	.
ALDH18A1	.
FLJ25404	-1.511110
JMJD2A	.
HGFAC	.
STX6	.
KIAA1505	.
GIPR	.
C10orf96	.
PRAMEF8	.
MAL	.
CGB2	.
APP	.
LGALS4	.
C1orf142	.
ASGR1	.
MST1	.
LCAT	.
AHSG	.
F2	.
CKAP2L	.
TTR	.
SSR1	.
SLC26A10	.
IL17RA	.
C14orf162	.
CKAP4	.
CTCFL	.
STAB1	.
HBG1	.
CALCOCO1	.
HIRA	.
IFRG15	.

FAM111B .
NHN1 .
TMPRSS11F .
OR5J2 .
SIGLEC10 .
ATF6 .
NLGN2 .
DPPA4 .
CCDC66 .
LIN28B .
SPATA13 .
RPL11 .
TRIM59 .
LRRIQ2 .
ITIH3 .
DDX6 .
LOC440248 .
PMS2CL 9.058008
P2RY4 .
FCGR3A .
RAG2 .
RSPO3 .
ZNF509 .
CAV3 .
ANXA6 .
SLC15A3 .
LARP7 .

(4) 이 data에 대해서는 Lasso method을 선택하는 것이 최적의 선택일 것이다. Ridge regression때의 람다 값이 Lasso method와 비교했을 때 매우 높게 측정된다. 이는, ridge regression을 통한 최적모델이 각 parameter들에 대한 계수들을 상당히 과소 추정한다는 것을 의미한다. 즉, 현재 설명변수들 중 반응변수와 관련이 없는 변수들이 많다는 것을 의미하며, 따라서 Lasso를 선택하여 해석력이 보다 좋은 모델을 selection하는 것이 유리할 것이다. 실제로 Lasso method의 경우 13개의 반응변수만을 채택하였다.