P4

November 20, 2021

[29]:
```
data_L <- read.table("table10.19.txt", head = TRUE)
```

[30]:
```
cor<-cor(data_L[2:7])
ev <- eigen(cor)
k_3 <- c()
for (i in 1:length(ev$values)) {
  evn <- ev$values[1:i]
  k_3[i] <- sqrt(max(evn)/min(evn))
}
names(k_3) <- c("k1","k2","k3", "k4", "k5", "k6")
k_3
```

k1 1 **k2** 1.97904844644675 **k3** 4.75702809548504 **k4** 17.5603715355781 **k5** 42.4709861933714 **k6** 110.544153442247

The correlation matrix shows some large coefficients between variables which indicates collinearity. There are six sets of variables, X1 and X2, X1 and X5, X1 and X6, X2 and X5, X2 and X6 and X5 and X6, which may exist collinearity. Then calculate the eigenvalues of correlation matrix and K values. The results show that $k4, k5$ and $k6$ are relatively large which means there exist three sets of collinearity. Equations are as follow: $\lambda_4 = 0.793\widetilde{X_1} - 0.122\widetilde{X_2} + 0.008\widetilde{X_3} - 0.077\widetilde{X_4} - 0.590\widetilde{X_5} - 0.052\widetilde{X_6}$ This equation can be simplified into: $0.122\widetilde{X_2} + 0.590\widetilde{X_5} = 0.793\widetilde{X_1}$$
$ X1, X2 and X5 are varaibles involved in this set of collinearity.

$\lambda_5 = -0.338\widetilde{X_1} + 0.150\widetilde{X_2} - 0.009\widetilde{X_3} - 0.024\widetilde{X_4} - 0.549\widetilde{X_5} + 0.750\widetilde{X_6}$ This equation can be simplified into: $0.549\widetilde{X_5} + 0.338\widetilde{X_1} = 0.150\widetilde{X_2} + 0.750\widetilde{X_6}$
$X1, X2, X5$ and $X6$ are varaibles involved in this set of collinearity.

$\lambda_6 = 0.135\widetilde{X_1} - 0.818\widetilde{X_2} - 0.107\widetilde{X_3} - 0.018\widetilde{X_4} + 0.312\widetilde{X_5} + 0.450\widetilde{X_6}$ This equation can be simplified into: $0.818\widetilde{X_2} + 0.107\widetilde{X_3} = 0.135\widetilde{X_1} + 0.312\widetilde{X_5} + 0.450\widetilde{X_6}$
$X1, X2, X3, X5$ and $X6$ are varaibles involved in this set of collinearity.

[31]:
```
lm <- lm(Y ~ X1+X2+X3+X4+X5+X6,data=data_L)
summary(lm)
```

```
Call:
lm(formula = Y ~ X1 + X2 + X3 + X4 + X5 + X6, data = data_L)

Residuals:
    Min      1Q  Median      3Q     Max
```

```
-410.11 -157.67  -28.16  101.55  455.39

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) -3.482e+06  8.904e+05  -3.911 0.003560 ** 
X1           1.506e+00  8.491e+00   0.177 0.863141    
X2          -3.582e-02  3.349e-02  -1.070 0.312681    
X3          -2.020e+00  4.884e-01  -4.136 0.002535 ** 
X4          -1.033e+00  2.143e-01  -4.822 0.000944 ***
X5          -5.110e-02  2.261e-01  -0.226 0.826212    
X6           1.829e+03  4.555e+02   4.016 0.003037 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 304.9 on 9 degrees of freedom
Multiple R-squared:  0.9955,	Adjusted R-squared:  0.9925 
F-statistic: 330.3 on 6 and 9 DF,  p-value: 4.984e-10
```

[32]:
```r
data_sc<-scale(data_L)
lm_sc <- lm(Y ~ X1+X2+X3+X4+X5+X6,data=as.data.frame(data_sc))
summary(lm_sc)
```

```
Call:
lm(formula = Y ~ X1 + X2 + X3 + X4 + X5 + X6, data = as.data.frame(data_sc))

Residuals:
      Min        1Q    Median        3Q       Max 
-0.116776 -0.044896 -0.008019  0.028916  0.129669 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.385e-17  2.170e-02   0.000 1.000000    
X1           4.628e-02  2.609e-01   0.177 0.863141    
X2          -1.014e+00  9.479e-01  -1.070 0.312681    
X3          -5.375e-01  1.300e-01  -4.136 0.002535 ** 
X4          -2.047e-01  4.246e-02  -4.822 0.000944 ***
X5          -1.012e-01  4.478e-01  -0.226 0.826212    
X6           2.480e+00  6.175e-01   4.016 0.003037 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.0868 on 9 degrees of freedom
Multiple R-squared:  0.9955,	Adjusted R-squared:  0.9925 
F-statistic: 330.3 on 6 and 9 DF,  p-value: 4.984e-10
```

```
[33]: lm_p <- princomp(~ X1 + X2 + X3 + X4 + X5 + X6,
                       data = as.data.frame(data_sc[,2:7]),
                       cor = TRUE)
      summary(lm_p,loadings = TRUE)
```

```
Importance of components:
                          Comp.1    Comp.2     Comp.3      Comp.4       Comp.5
Standard deviation     2.1455482 1.0841312 0.45102702 0.122181253 0.0505179747
Proportion of Variance 0.7672295 0.1958901 0.03390423 0.002488043 0.0004253443
Cumulative Proportion  0.7672295 0.9631196 0.99702383 0.999511871 0.9999372153
                             Comp.6
Standard deviation     1.940897e-02
Proportion of Variance 6.278469e-05
Cumulative Proportion  1.000000e+00

Loadings:
   Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6
X1  0.462         0.149  0.793  0.338  0.135
X2  0.462         0.278 -0.122 -0.150 -0.818
X3  0.321 -0.596 -0.728               -0.107
X4  0.202  0.798 -0.562
X5  0.462         0.196 -0.590  0.549  0.312
X6  0.465         0.128        -0.750  0.450
```

```
[34]: lm_p$loadings[1,]
```

Comp.1 0.461834898166772 **Comp.2** 0.0578427676677562 **Comp.3** 0.149119892053081 **Comp.4** 0.792873558903563 **Comp.5** 0.337937826207117 **Comp.6** 0.135187070638823

```
[35]: predict(lm_p)
```

A matrix: 16×6 of type dbl

	Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6
1	-3.59294203	-0.7761110	0.31804507	-0.169624216	0.009085721	0.00266
2	-3.10924187	-0.8768853	0.66329350	0.130051755	0.063565251	0.01237
3	-2.42014988	-1.5905016	-0.50961596	-0.009106066	0.005934717	0.00522
4	-2.16257276	-1.3181781	-0.11494311	-0.063265342	-0.063873551	-0.0141
5	-1.48540767	1.2763227	-0.03004947	0.100660376	0.053969833	-0.0440
6	-1.04339031	1.9851409	-0.16650384	0.047914580	0.038252420	-0.0131
7	-0.72546382	1.9732212	0.06933769	-0.073217347	0.022425386	0.03281
8	0.03355589	0.6124972	-1.07307182	-0.067293112	-0.002331281	0.01724
9	0.10277563	0.7162362	-0.10076710	-0.104426569	-0.102049470	-0.0195
10	0.46417097	0.5658113	0.30255566	0.018137849	-0.086508832	0.01460
11	0.98638447	0.4435320	0.45984009	0.123243732	-0.024470311	0.02804
12	1.87669233	-0.8914800	-0.69963411	0.193195372	0.022381351	0.00836
13	2.00361238	-0.3992485	0.27468475	0.148640632	-0.037890240	-0.0243
14	2.43855615	-0.5154723	0.37766452	0.063619308	-0.016767727	0.00450
15	3.17896368	-1.0224220	-0.20859150	-0.070338720	0.058280241	-0.0013
16	3.45445683	-0.1824626	0.43775561	-0.268192230	0.059996493	-0.0092

[36]:
```
d<-cbind(scale(data_L$Y),predict(lm_p))
colnames(d)<-c("Y","Z1","Z2","Z3","Z4","Z5","Z6")
lm_pr<-lm(Y ~ Z1 + Z2 + Z3 + Z4 + Z5 + Z6,
          data = as.data.frame(d))
summary(lm_pr)
```

```
Call:
lm(formula = Y ~ Z1 + Z2 + Z3 + Z4 + Z5 + Z6, data = as.data.frame(d))

Residuals:
      Min        1Q    Median        3Q       Max 
-0.116776 -0.044896 -0.008019  0.028916  0.129669 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.043e-16  2.170e-02   0.000 1.000000    
Z1           4.315e-01  1.011e-02  42.662 1.07e-11 ***
Z2           1.080e-01  2.002e-02   5.397 0.000435 ***
Z3           5.129e-01  4.811e-02  10.660 2.10e-06 ***
Z4           9.851e-02  1.776e-01   0.555 0.592672    
Z5          -1.701e+00  4.296e-01  -3.960 0.003305 ** 
Z6           1.920e+00  1.118e+00   1.717 0.120105    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.0868 on 9 degrees of freedom
Multiple R-squared:  0.9955,	Adjusted R-squared:  0.9925 
F-statistic: 330.3 on 6 and 9 DF,  p-value: 4.984e-10
```

[37]:
```
lm_pr$coefficients
```

(Intercept) 1.04279817132584e-16 **Z1** 0.431499810989923 **Z2** 0.108026487988878 **Z3** 0.512906534728037 **Z4** 0.0985071028187975 **Z5** -1.70101494756945 **Z6** 1.91979308127566

[38]:
```
theta<-c()
for (i in 1:6) {
  theta[i]<-t(as.matrix(lm_p$loadings[i,])) %*% as.matrix(lm_pr$coefficients[2:
 ↪7])
}
names(theta)<-c("theta1","theta2","theta3","theta4","theta5","theta6")
theta
```

theta1 0.0448123757583045 **theta2** -0.98155568146248 **theta3** -0.520473362768668 **theta4** -0.198239322935066 **theta5** -0.0980069221492109 **theta6** 2.40092471483419

The θ_j are as above. According to the cumulative proportion, the Comp. 1 and Comp. 2 will be chosen to construct the model, because the cumulative proportion of the two components is larger than 95%. The estimates of the coefficients in the standardized model are as above.

[39]:
```r
summary(lm_p,loadings=TRUE)
```

```
Importance of components:
                          Comp.1    Comp.2     Comp.3      Comp.4       Comp.5
Standard deviation     2.1455482 1.0841312 0.45102702 0.122181253 0.0505179747
Proportion of Variance 0.7672295 0.1958901 0.03390423 0.002488043 0.0004253443
Cumulative Proportion  0.7672295 0.9631196 0.99702383 0.999511871 0.9999372153
                             Comp.6
Standard deviation     1.940897e-02
Proportion of Variance 6.278469e-05
Cumulative Proportion  1.000000e+00

Loadings:
   Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6
X1  0.462         0.149  0.793  0.338  0.135
X2  0.462         0.278 -0.122 -0.150 -0.818
X3  0.321 -0.596 -0.728               -0.107
X4  0.202  0.798 -0.562
X5  0.462         0.196 -0.590  0.549  0.312
X6  0.465         0.128        -0.750  0.450
```

[40]:
```r
pre<-predict(lm_p)
data_L$z1 <-pre[,1]
data_L$z2 <-pre[,2]
data_L$y <-scale(data_L$Y)
lm_pr<-lm(y~ z1 + z2,data = data_L)
summary(lm_pr)
```

```
Call:
lm(formula = y ~ z1 + z2, data = data_L)

Residuals:
     Min       1Q   Median       3Q      Max
-0.52370 -0.14925  0.01296  0.21234  0.46183

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 8.252e-17  7.161e-02   0.000    1.000
z1          4.315e-01  3.338e-02  12.928 8.51e-09 ***
z2          1.080e-01  6.606e-02   1.635    0.126
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2865 on 13 degrees of freedom
```

```
Multiple R-squared:  0.9289,	Adjusted R-squared:  0.9179
F-statistic:  84.9 on 2 and 13 DF,  p-value: 3.45e-08
```

[41]:
```
as.matrix(c(lm_pr$coefficients[2:3],0,0,0,0))
```

A matrix: 6 × 1 of type dbl

z1	0.4314998
z2	0.1080265
	0.0000000
	0.0000000
	0.0000000
	0.0000000

[42]:
```
theta_n<-c()
for (i in 1:6) {
  theta_n[i]<-t(as.matrix(lm_p$loadings[i,]))%*%as.
↪matrix(c(lm_pr$coefficients[2:3],0,0,0,0))
}
names(theta_n)<-c("theta1","theta2","theta3","theta4","theta5","theta6")
theta_n
```

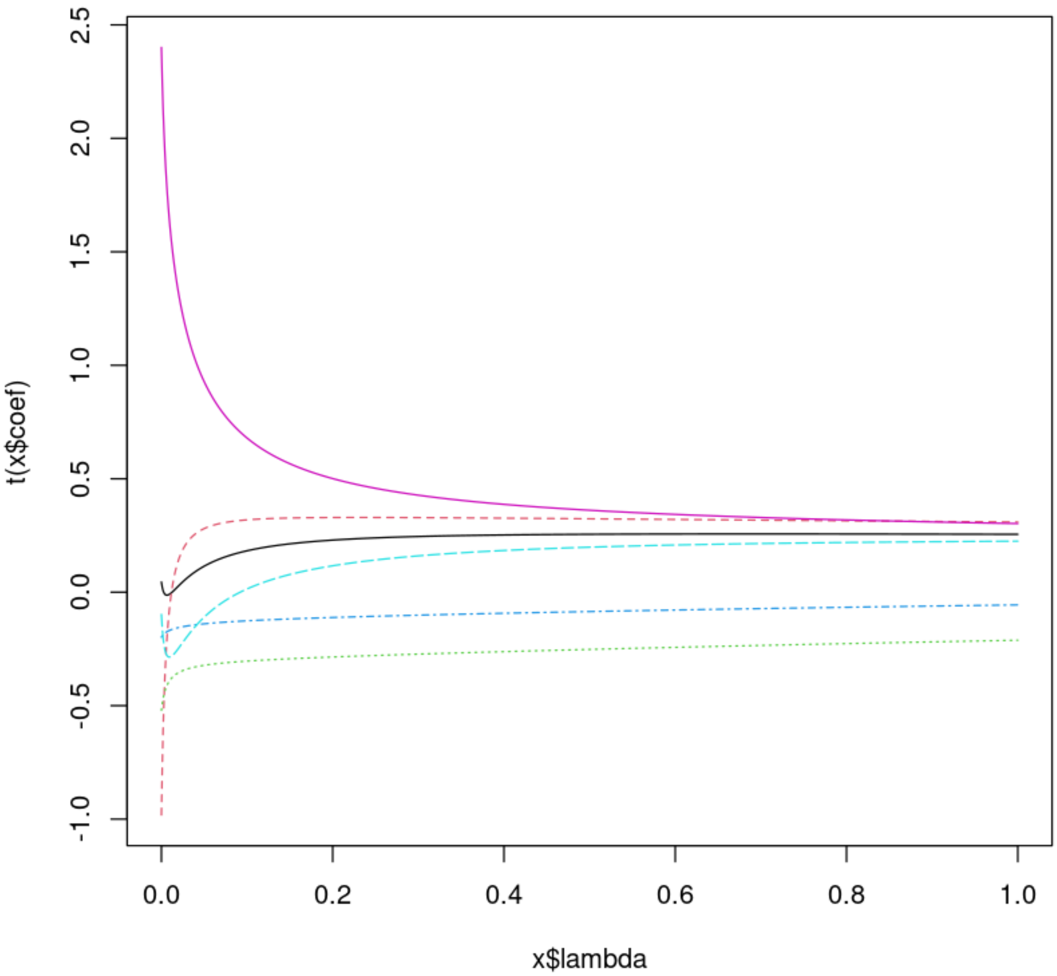
theta1 0.205530222314217 **theta2** 0.204887374762976 **theta3** 0.0743168342522829 **theta4** 0.173177353225546 **theta5** 0.194553444341986 **theta6** 0.200688490305094

[43]:
```
library(MASS)
library("car")
```

[44]:
```
lm_r<- lm.ridge(Y ~ X1 + X2 + X3 + X4 + X5 + X6,
                data = as.data.frame(data_sc),
                 lambda = seq(0,1,0.001))
select(lm_r)
```

```
modified HKB estimator is 0.004275357
modified L-W estimator is 0.03229531
smallest value of GCV  at 0.003
```

[45]:
```
plot(lm_r)
```



[46]:
```
vif(lm(Y ~ X1 + X2 + X3 + X4 + X5 + X6,
       data = as.data.frame(data_sc)))
```

X1 135.532438280047 **X2** 1788.51348271876 **X3** 33.6188905960544 **X4** 3.58893019344586 **X5** 399.151022312768 **X6** 758.980597407142

As shown above, the recommended value for k is 0.003.

[47]:
```
lm.ridge(Y ~ X1 + X2 + X3 + X4 + X5 + X6,
         data = as.data.frame(data_sc),
         lambda = 0.003)
```

```
                          X1            X2            X3            X4
-6.253445e-18 -3.244788e-03 -4.924111e-01 -4.652798e-01 -1.896297e-01
```

```
            X5             X6
-2.399194e-01  2.092754e+00
```

The estimates of the regression coefficients θ_j of the standardized model are as above.