Hw5_401

Hongkai Lou

2024-11-05

```r
sigma = matrix(0.9, nrow=4, ncol=4) + .1*diag(4)
A = chol(sigma)
print(A)
```

```
##      [,1]      [,2]      [,3]      [,4]
## [1,]    1 0.9000000 0.9000000 0.9000000
## [2,]    0 0.4358899 0.2064742 0.2064742
## [3,]    0 0.0000000 0.3838859 0.1233919
## [4,]    0 0.0000000 0.0000000 0.3635146
```

```r
t(A) %*% A
```

```
##      [,1] [,2] [,3] [,4]
## [1,]  1.0  0.9  0.9  0.9
## [2,]  0.9  1.0  0.9  0.9
## [3,]  0.9  0.9  1.0  0.9
## [4,]  0.9  0.9  0.9  1.0
```

3(b): We see the issue here there are diagonal elements larger than 1. Thus, although it approximately equal the assumed covariance matrix, there is certain difference.

```r
set.seed(89)
Z = matrix(rnorm(4000), nrow=1000)
X = Z %*% A
var(X)
```

```
##           [,1]      [,2]      [,3]      [,4]
## [1,] 1.0511258 0.9183115 0.9515287 0.9491976
## [2,] 0.9183115 0.9976135 0.9262391 0.9209018
## [3,] 0.9515287 0.9262391 1.0417162 0.9509703
## [4,] 0.9491976 0.9209018 0.9509703 1.0454256
```

3(c)

```r
set.seed(12345)
sigma = matrix(0.9, nrow=15, ncol=15) + .1*diag(15)
A = chol(sigma)
Z = matrix(rnorm(10100*15), nrow=10100)
X = Z %*% A
# generate a new Z, A and X
beta = c(1,-1,1.5,0.5,-0.5,rep(0,10))
e = rnorm(10100)*3
y = 3 + X %*% beta + e
```

3(d)

```r
dat = data.frame(X)
dat$y <- y
train <- c(rep(T,100), rep(F, 10000))
test <- dat[c(!train),]
lm_ols <- lm(y ~ X1+X2+X3+X4+X5, data = dat[train,])
summary(lm_ols)
```

```
##
## Call:
## lm(formula = y ~ X1 + X2 + X3 + X4 + X5, data = dat[train, ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.8436 -2.0442  0.2997  1.8333  6.9526
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0256     0.3295   9.183    1e-14 ***
## X1            0.9439     0.8875   1.064  0.29026
## X2           -1.6256     1.0049  -1.618  0.10906
## X3            2.7879     0.8924   3.124  0.00237 **
## X4           -0.3034     1.0439  -0.291  0.77200
## X5           -0.3711     0.8164  -0.455  0.65048
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.2 on 94 degrees of freedom
## Multiple R-squared:  0.2407, Adjusted R-squared:  0.2003
## F-statistic: 5.961 on 5 and 94 DF,  p-value: 7.843e-05
```

The Residual Standard Error is 3.398, with 94 degrees of freedom, we estimate σ_e^2 with the formula $3.398^2 = 11.5464$. The slope are $B_1 = 0.9439$, $B_2 = -1.6256$, $B_3 = 2.7879$, $B_4 = -0.3034$, $B_5 = -0.3711$ respectively. The R^2 value is 0.2101. All the estimate are within 2 standard error of the true estimate, though one possible reason is the standard errors are too large.

• The slope have correct signs. Only B_2 is significant and will pass the hypothesis test at a significance level of p = 0.05. The 95 confidence intervals as follow covers the true value. The slope of B_4 is not in the correct sign

```r
confint(lm_ols)
```

```
##                  2.5 %    97.5 %
## (Intercept)  2.3714294 3.6797538
## X1          -0.8182307 2.7059553
## X2          -3.6208313 0.3695436
## X3           1.0160970 4.5597365
## X4          -2.3759925 1.7692921
## X5          -1.9919717 1.2498126
```

3(e): The MSE is 9.4495

```r
mean((test$y-predict(lm_ols, test))^2)
```

```
## [1] 9.449501
```

3(f): The coefficient are within two standard errors of their true values. x1 to x5 have the correct sign. None of them are actually significant.

```r
lm_allols <- lm(y~., data = dat[train,])
summary(lm_allols)
```

```
##
## Call:
## lm(formula = y ~ ., data = dat[train, ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.7768 -1.8727  0.0985  1.8531  6.4236
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.09711    0.34378   9.009 5.69e-14 ***
## X1           1.64535    1.01410   1.622  0.10845
## X2          -1.27455    1.12632  -1.132  0.26102
## X3           3.04446    0.99629   3.056  0.00301 **
## X4           0.17894    1.16865   0.153  0.87867
## X5           0.12057    0.95410   0.126  0.89974
## X6           0.42167    1.04928   0.402  0.68880
## X7          -0.05058    1.16496  -0.043  0.96547
## X8          -1.48874    1.18517  -1.256  0.21255
## X9           1.02701    1.03928   0.988  0.32589
## X10         -0.83981    1.13596  -0.739  0.46179
## X11          0.68516    1.02798   0.667  0.50691
## X12         -0.55163    1.07908  -0.511  0.61055
## X13         -1.25600    1.22391  -1.026  0.30773
## X14          0.52319    1.01348   0.516  0.60705
## X15         -0.73817    1.23259  -0.599  0.55086
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.258 on 84 degrees of freedom
## Multiple R-squared:  0.2964, Adjusted R-squared:  0.1708
## F-statistic: 2.359 on 15 and 84 DF,  p-value: 0.007036
```

```r
print(lm_allols$coefficients)
```

```
## (Intercept)          X1          X2          X3          X4          X5
##  3.09711359  1.64535077 -1.27455388  3.04445854  0.17893846  0.12057058
##          X6          X7          X8          X9         X10         X11
##  0.42167092 -0.05058088 -1.48874409  1.02701372 -0.83981022  0.68515562
##         X12         X13         X14         X15
## -0.55162784 -1.25600264  0.52318748 -0.73817272
```

```r
print(summary(lm_allols)$sigma^2)
```

```
## [1] 10.61713
```

```r
print(summary(lm_allols)$r.squared)
```

```
## [1] 0.2964292
```

3(g): The MSE value is 10.23477. We see that the value of MSE increases this time compared to previous one.

```r
mean((test$y-predict(lm_allols, test))^2)
```

```
## [1] 10.23477
```

3(h): We observe from the final model that not all of the true variable are included. Only X1 and X3 are included in the model at the end of backward selection.

```r
lm_stepwise <- lm(y~., data = dat[train,])
step_fit <- step(lm_stepwise, test = 'F')
```

```
## Start:  AIC=250.81
## y ~ X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8 + X9 + X10 + X11 +
##     X12 + X13 + X14 + X15
##
##        Df Sum of Sq    RSS    AIC F value   Pr(>F)
## - X7    1     0.020 891.86 248.81  0.0019 0.965471
## - X5    1     0.170 892.01 248.83  0.0160 0.899740
## - X4    1     0.249 892.09 248.84  0.0234 0.878674
## - X6    1     1.715 893.55 249.00  0.1615 0.688804
## - X12   1     2.775 894.61 249.12  0.2613 0.610549
## - X14   1     2.829 894.67 249.13  0.2665 0.607051
## - X15   1     3.808 895.65 249.24  0.3587 0.550864
## - X11   1     4.716 896.56 249.34  0.4442 0.506913
## - X10   1     5.803 897.64 249.46  0.5466 0.461790
## - X9    1    10.368 902.21 249.97  0.9765 0.325893
## - X13   1    11.181 903.02 250.06  1.0531 0.307732
## - X2    1    13.596 905.43 250.32  1.2805 0.261021
## - X8    1    16.753 908.59 250.67  1.5779 0.212548
## <none>              891.84 250.81
## - X1    1    27.949 919.79 251.90  2.6324 0.108449
## - X3    1    99.141 990.98 259.35  9.3378 0.003009 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=248.81
## y ~ X1 + X2 + X3 + X4 + X5 + X6 + X8 + X9 + X10 + X11 + X12 +
##     X13 + X14 + X15
##
##        Df Sum of Sq    RSS    AIC F value   Pr(>F)
## - X5    1     0.152 892.01 246.83  0.0145 0.904471
## - X4    1     0.231 892.09 246.84  0.0220 0.882422
## - X6    1     1.734 893.59 247.01  0.1653 0.685355
## - X12   1     2.756 894.61 247.12  0.2626 0.609641
## - X14   1     2.827 894.69 247.13  0.2694 0.605067
## - X15   1     3.861 895.72 247.25  0.3680 0.545711
## - X11   1     4.724 896.58 247.34  0.4502 0.504046
## - X10   1     5.882 897.74 247.47  0.5606 0.456071
## - X9    1    10.397 902.26 247.97  0.9909 0.322339
## - X13   1    11.272 903.13 248.07  1.0743 0.302912
## - X2    1    13.629 905.49 248.33  1.2989 0.257609
## <none>              891.86 248.81
## - X8    1    18.293 910.15 248.84  1.7434 0.190247
## - X1    1    28.016 919.87 249.91  2.6701 0.105949
## - X3    1    99.609 991.47 257.40  9.4934 0.002779 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Step:  AIC=246.83
## y ~ X1 + X2 + X3 + X4 + X6 + X8 + X9 + X10 + X11 + X12 + X13 +
##     X14 + X15
##
##        Df Sum of Sq    RSS    AIC F value   Pr(>F)
## - X4    1     0.278 892.29 244.86  0.0268 0.870284
## - X6    1     1.744 893.76 245.03  0.1682 0.682754
## - X12   1     2.645 894.66 245.13  0.2550 0.614855
## - X14   1     2.856 894.87 245.15  0.2753 0.601135
## - X15   1     3.761 895.77 245.25  0.3626 0.548636
## - X11   1     4.683 896.69 245.35  0.4515 0.503412
## - X10   1     5.741 897.75 245.47  0.5535 0.458933
## - X9    1    11.093 903.10 246.07  1.0695 0.303953
## - X13   1    11.266 903.28 246.09  1.0862 0.300232
## - X2    1    13.485 905.50 246.33  1.3002 0.257350
## <none>              892.01 246.83
## - X8    1    18.160 910.17 246.85  1.7508 0.189281
## - X1    1    28.105 920.12 247.93  2.7096 0.103393
## - X3    1    99.908 991.92 255.45  9.6323 0.002588 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=244.86
## y ~ X1 + X2 + X3 + X6 + X8 + X9 + X10 + X11 + X12 + X13 + X14 +
##     X15
##
##        Df Sum of Sq     RSS    AIC F value   Pr(>F)
## - X6    1     1.657  893.95 243.05  0.1615 0.688748
## - X12   1     2.416  894.70 243.13  0.2355 0.628681
## - X14   1     3.064  895.35 243.21  0.2988 0.586062
## - X15   1     3.975  896.26 243.31  0.3876 0.535199
## - X11   1     4.841  897.13 243.40  0.4720 0.493898
## - X10   1     5.466  897.75 243.47  0.5329 0.467351
## - X9    1    10.815  903.10 244.07  1.0545 0.307320
## - X13   1    10.991  903.28 244.09  1.0717 0.303442
## - X2    1    13.261  905.55 244.34  1.2929 0.258627
## <none>               892.29 244.86
## - X8    1    18.594  910.88 244.92  1.8130 0.181648
## - X1    1    30.687  922.98 246.24  2.9920 0.087221 .
## - X3    1   108.286 1000.58 254.32 10.5581 0.001645 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=243.05
## y ~ X1 + X2 + X3 + X8 + X9 + X10 + X11 + X12 + X13 + X14 + X15
##
##        Df Sum of Sq     RSS    AIC F value   Pr(>F)
## - X12   1     1.931  895.88 241.26  0.1901 0.663904
## - X14   1     3.376  897.32 241.42  0.3323 0.565761
## - X15   1     3.575  897.52 241.45  0.3519 0.554551
## - X10   1     5.166  899.11 241.62  0.5085 0.477660
## - X11   1     5.310  899.26 241.64  0.5227 0.471619
## - X13   1    10.163  904.11 242.18  1.0005 0.319934
## - X9    1    13.007  906.95 242.49  1.2804 0.260899
```

```
## - X2    1    13.727  907.67 242.57  1.3513 0.248191
## - X8    1    17.124  911.07 242.94  1.6857 0.197566
## <none>              893.95 243.05
## - X1    1    30.104  924.05 244.36  2.9634 0.088681 .
## - X3    1   112.298 1006.24 252.88 11.0546 0.001291 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=241.26
## y ~ X1 + X2 + X3 + X8 + X9 + X10 + X11 + X13 + X14 + X15
##
##        Df Sum of Sq     RSS    AIC F value   Pr(>F)
## - X14   1     2.655  898.53 239.56  0.2638 0.608799
## - X11   1     4.335  900.21 239.75  0.4307 0.513362
## - X15   1     5.561  901.44 239.88  0.5525 0.459266
## - X10   1     5.975  901.85 239.93  0.5936 0.443084
## - X13   1    10.383  906.26 240.42  1.0315 0.312559
## - X9    1    12.042  907.92 240.60  1.1963 0.277014
## - X2    1    14.287  910.16 240.84  1.4193 0.236688
## <none>              895.88 241.26
## - X8    1    18.184  914.06 241.27  1.8064 0.182351
## - X1    1    28.913  924.79 242.44  2.8724 0.093609 .
## - X3    1   114.969 1010.85 251.34 11.4215 0.001079 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=239.56
## y ~ X1 + X2 + X3 + X8 + X9 + X10 + X11 + X13 + X15
##
##        Df Sum of Sq     RSS    AIC F value    Pr(>F)
## - X15   1     3.903  902.43 237.99  0.3909 0.5334029
## - X11   1     4.080  902.61 238.01  0.4087 0.5242478
## - X10   1     6.139  904.67 238.24  0.6149 0.4350207
## - X13   1    11.402  909.93 238.82  1.1421 0.2880714
## - X2    1    12.260  910.79 238.91  1.2280 0.2707477
## - X9    1    13.689  912.22 239.07  1.3712 0.2447025
## - X8    1    16.903  915.43 239.42  1.6930 0.1965231
## <none>              898.53 239.56
## - X1    1    32.553  931.08 241.12  3.2606 0.0743072 .
## - X3    1   126.019 1024.55 250.68 12.6225 0.0006092 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=237.99
## y ~ X1 + X2 + X3 + X8 + X9 + X10 + X11 + X13
##
##        Df Sum of Sq     RSS    AIC F value   Pr(>F)
## - X11   1     4.159  906.59 236.45  0.4194 0.518892
## - X10   1     9.844  912.28 237.08  0.9926 0.321742
## - X9    1    12.423  914.86 237.36  1.2527 0.265986
## - X2    1    13.646  916.08 237.49  1.3761 0.243835
## - X13   1    15.335  917.77 237.68  1.5464 0.216866
## <none>              902.43 237.99
## - X8    1    18.610  921.04 238.03  1.8766 0.174096
```

```
## - X1    1    29.397  931.83 239.20  2.9644 0.088515 .
## - X3    1   124.400 1026.84 248.91 12.5443 0.000629 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=236.45
## y ~ X1 + X2 + X3 + X8 + X9 + X10 + X13
##
##        Df Sum of Sq     RSS    AIC F value    Pr(>F)
## - X10   1     8.485  915.08 235.38  0.8611 0.3558650
## - X2    1    10.845  917.44 235.64  1.1005 0.2968965
## - X13   1    13.060  919.65 235.88  1.3253 0.2526220
## - X9    1    15.569  922.16 236.16  1.5800 0.2119480
## - X8    1    16.693  923.29 236.28  1.6940 0.1963278
## <none>               906.59 236.45
## - X1    1    30.470  937.06 237.76  3.0921 0.0819985 .
## - X3    1   125.426 1032.02 247.41 12.7280 0.0005744 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=235.38
## y ~ X1 + X2 + X3 + X8 + X9 + X13
##
##        Df Sum of Sq     RSS    AIC F value    Pr(>F)
## - X9    1    10.207  925.29 234.49  1.0373 0.3110909
## - X2    1    14.067  929.15 234.91  1.4297 0.2348574
## <none>               915.08 235.38
## - X13   1    18.690  933.77 235.41  1.8995 0.1714441
## - X8    1    20.014  935.09 235.55  2.0340 0.1571597
## - X1    1    29.017  944.10 236.51  2.9491 0.0892565 .
## - X3    1   117.830 1032.91 245.50 11.9751 0.0008159 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=234.49
## y ~ X1 + X2 + X3 + X8 + X13
##
##        Df Sum of Sq     RSS    AIC F value    Pr(>F)
## - X2    1    11.643  936.93 233.74  1.1829 0.2795567
## - X13   1    13.178  938.46 233.91  1.3388 0.2501819
## - X8    1    15.369  940.65 234.14  1.5614 0.2145642
## <none>               925.29 234.49
## - X1    1    32.185  957.47 235.91  3.2697 0.0737703 .
## - X3    1   137.543 1062.83 246.35 13.9730 0.0003184 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=233.74
## y ~ X1 + X3 + X8 + X13
##
##        Df Sum of Sq     RSS    AIC F value    Pr(>F)
## <none>               936.93 233.74
## - X13   1    19.857  956.79 233.84  2.0134 0.1591849
## - X1    1    24.690  961.62 234.34  2.5034 0.1169212
```

```
## - X8    1    25.377  962.31 234.42  2.5731 0.1120111
## - X3    1   126.212 1063.14 244.38 12.7972 0.0005484 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
lm_backstep <- lm(y~X1+X3+X8+X13, data = dat[train,])
summary(lm_backstep)
```

```
##
## Call:
## lm(formula = y ~ X1 + X3 + X8 + X13, data = dat[train, ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.2310 -1.8975  0.2254  1.6861  7.4489
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0673     0.3217   9.533 1.64e-15 ***
## X1            1.3978     0.8835   1.582 0.116921
## X3            3.0285     0.8466   3.577 0.000548 ***
## X8           -1.5716     0.9797  -1.604 0.112011
## X13          -1.4510     1.0226  -1.419 0.159185
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.14 on 95 degrees of freedom
## Multiple R-squared:  0.2609, Adjusted R-squared:  0.2297
## F-statistic: 8.382 on 4 and 95 DF,  p-value: 7.787e-06
```

```
print(mean((test$y-predict(lm_backstep, test))^2))
```

```
## [1] 10.04426
```

Now we perform forward selection

```
lm1 = lm(y~1, data = dat[train,])
lm_forward <- step(lm1, scope=~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10+X11+X12+X13+X14+X15,  test="F")
```

```
## Start:  AIC=255.97
## y ~ 1
##
##        Df Sum of Sq    RSS    AIC F value    Pr(>F)
## + X3    1    256.55 1011.0 235.36  24.868 2.655e-06 ***
## + X1    1    195.03 1072.6 241.26  17.820 5.434e-05 ***
## + X9    1    189.14 1078.5 241.81  17.187 7.204e-05 ***
## + X14   1    182.34 1085.2 242.44  16.465 9.965e-05 ***
## + X4    1    175.79 1091.8 243.04  15.779 0.0001360 ***
## + X6    1    170.85 1096.7 243.49  15.267 0.0001718 ***
## + X11   1    168.33 1099.3 243.72  15.007 0.0001935 ***
## + X7    1    166.22 1101.4 243.91  14.790 0.0002138 ***
## + X5    1    158.74 1108.8 244.59  14.030 0.0003039 ***
## + X10   1    150.15 1117.4 245.36  13.169 0.0004545 ***
## + X13   1    149.97 1117.6 245.38  13.150 0.0004585 ***
## + X12   1    146.97 1120.6 245.65  12.853 0.0005274 ***
## + X15   1    146.37 1121.2 245.70  12.793 0.0005424 ***
## + X2    1    146.04 1121.5 245.73  12.761 0.0005509 ***
```

```
## + X8    1    140.49 1127.1 246.22  12.215 0.0007135 ***
## <none>              1267.6 255.97
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=235.36
## y ~ X3
##
##        Df Sum of Sq     RSS    AIC F value    Pr(>F)
## + X8    1    41.330  969.71 233.18  4.1343   0.04475 *
## + X13   1    37.991  973.05 233.53  3.7872   0.05454 .
## + X2    1    36.449  974.59 233.68  3.6277   0.05979 .
## + X10   1    32.480  978.56 234.09  3.2196   0.07587 .
## + X15   1    31.377  979.66 234.20  3.1068   0.08112 .
## + X12   1    24.221  986.82 234.93  2.3808   0.12609
## + X7    1    20.373  990.66 235.32  1.9948   0.16104
## <none>              1011.04 235.36
## + X5    1    14.192  996.84 235.94  1.3810   0.24281
## + X4    1    12.596  998.44 236.10  1.2237   0.27138
## + X6    1    10.300 1000.74 236.33  0.9984   0.32019
## + X11   1     8.988 1002.05 236.46  0.8700   0.35327
## + X14   1     5.646 1005.39 236.80  0.5447   0.46227
## + X9    1     2.311 1008.73 237.13  0.2222   0.63840
## + X1    1     1.146 1009.89 237.24  0.1101   0.74080
## - X3    1   256.553 1267.59 255.97 24.8678 2.655e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=233.18
## y ~ X3 + X8
##
##        Df Sum of Sq     RSS    AIC F value   Pr(>F)
## <none>               969.71 233.18
## + X1    1    12.920  956.79 233.84  1.2963 0.257719
## + X13   1     8.087  961.62 234.34  0.8074 0.371152
## + X2    1     7.928  961.78 234.36  0.7914 0.375912
## + X10   1     7.082  962.62 234.45  0.7063 0.402769
## + X15   1     5.434  964.27 234.62  0.5410 0.463832
## + X9    1     4.817  964.89 234.68  0.4792 0.490442
## + X12   1     1.821  967.89 234.99  0.1806 0.671797
## + X14   1     1.283  968.42 235.05  0.1272 0.722167
## + X11   1     0.605  969.10 235.12  0.0600 0.807084
## + X4    1     0.533  969.17 235.13  0.0528 0.818752
## + X6    1     0.518  969.19 235.13  0.0513 0.821360
## + X7    1     0.313  969.39 235.15  0.0310 0.860611
## + X5    1     0.085  969.62 235.17  0.0084 0.927101
## - X8    1    41.330 1011.04 235.36  4.1343 0.044755 *
## - X3    1   157.396 1127.10 246.22 15.7443 0.000139 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We see that the MSE value of backward selection is 10.0443, which performs better than the OLS model with the full model. However, it's still large compared to the MSE value of 9.449501 of the first OLS model with the true parameters only. The forward selection has a MSE value of 9.8797, which is less than the MSE value of backward selection.

```r
lm_forward = lm(y~X3+X8, data = dat[train,])
summary(lm_forward)
```

```
##
## Call:
## lm(formula = y ~ X3 + X8, data = dat[train, ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.7523 -2.1644  0.2534  1.8345  7.6125
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.1062     0.3229   9.618 8.95e-16 ***
## X3            2.9406     0.7411   3.968 0.000139 ***
## X8           -1.4696     0.7228  -2.033 0.044755 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.162 on 97 degrees of freedom
## Multiple R-squared:  0.235,  Adjusted R-squared:  0.2192
## F-statistic:  14.9 on 2 and 97 DF,  p-value: 2.278e-06
```

```r
print(mean((test$y-predict(lm_forward, test))^2))
```

```
## [1] 9.879715
```

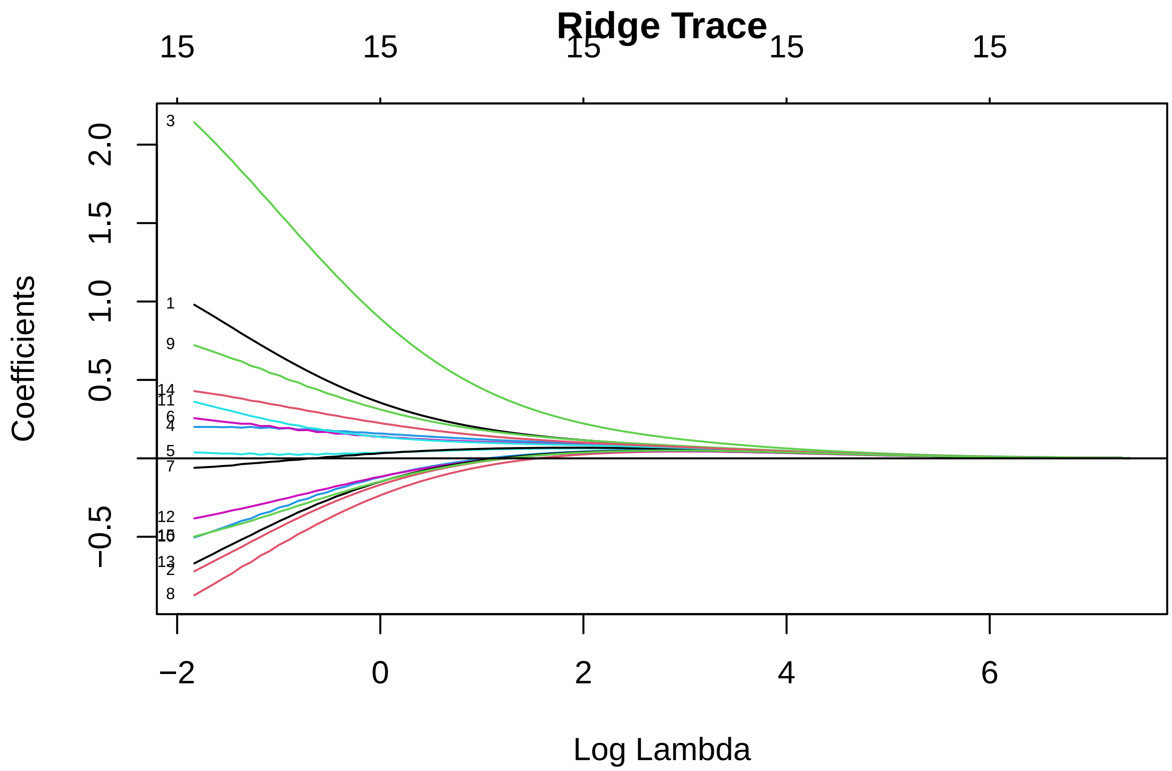
3(j)

```r
library(glmnet)
```

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

```
## Loaded glmnet 4.1-8
```

```r
x_train = model.matrix(y~.-1, dat[train,])
fit_ridge <- cv.glmnet(x_train, dat[train,]$y, alpha = 0)
plot(fit_ridge$glmnet.fit, "lambda", label=TRUE)
abline(h = 0)
title('Ridge Trace')
```



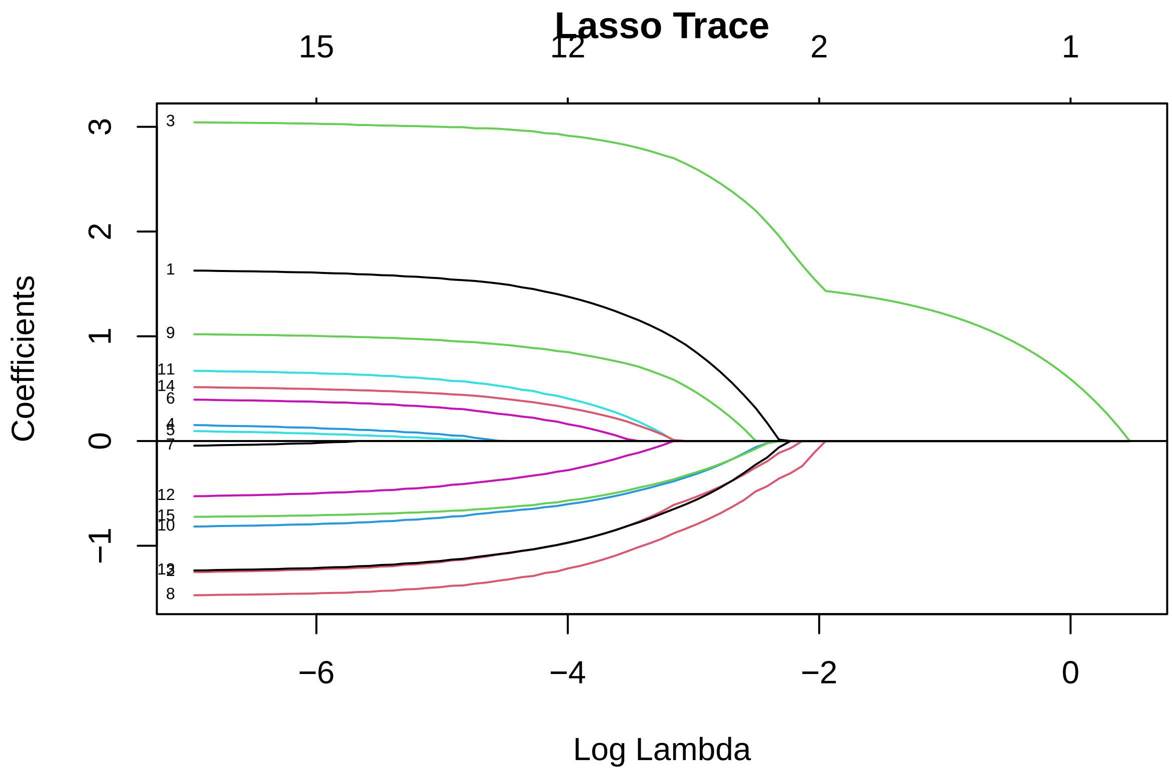
3(k): We see the MSE value is relatively small at 9.57, which is very close to the OLS model with true parameters included only. It proves the ability of ridge regression on penalize regression terms and shrink the variables.

```
x_test = model.matrix(y~.-1, test)
prediction <- predict(fit_ridge, s = fit_ridge$lambda.min, newx = x_test)
mean((test$y-prediction)^2)
```

```
## [1] 9.570036
```

3(I)

```
fit_lasso <- cv.glmnet(x_train, dat[train,]$y, alpha = 1)
plot(fit_lasso$glmnet.fit, 'lambda', label = T)
abline(h = 0)
title('Lasso Trace')
```




```
prediction <- predict(fit_lasso, s = fit_lasso$lambda.min, newx = x_test)
mean((test$y-prediction)^2)
```

```
## [1] 9.41732
```

The MSE value of the lasso regression is even smaller than the true ols model with the 5 true parameter. The lasso regression has the ability to set estimate coefficient exactly to 0, which is helpful in eliminating x6-x15 in this specific case.

```
hw5 <- function(beta=c(1,-1,1.5,0.5,-0.5,rep(0,10)), rho=0.9, sigmae=3, seed=12345, ntrain=100, ntest=1(
{
  set.seed(seed)
  n = ntrain + ntest
  Z <- matrix(rnorm(n*15), nrow=n)
  e <- rnorm(n)*sigmae
  sigma <- matrix(rho, nrow=15, ncol=15) + diag(rep(1-rho, 15))
  A <- chol(sigma)
  X <- Z %*% A
  y <- 3 + X %*% beta + e
  train <- data.frame(X[1:ntrain,], y=y[1:ntrain])
  test <- data.frame(X[(ntrain+1):n,], y=y[(ntrain+1):n])
  cat("-------------------------------------------\n")
  cat("correlation between x: ", rho, "\n")
  cat("Error variance: ", sigmae^2, "\n")

  # OLS of x1-x5 as predictors
  fit <- lm(y~X1+X2+X3+X4+X5, train)
  print(summary(fit))
  cat("OLS x1-x5:", mean((test$y-predict(fit,test))^2), "\n")

  # fit OLS model with x1-x15
```

```r
  fit <- lm(y~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10+X11+X12+X13+X14+X15, train)
  cat("OLS x1-x15:", mean((test$y-predict(fit,test))^2), "\n")

  # backward
  fit2 <- step(fit, scope=~1, trace=F)
  cat("backward (", fit2$rank-1, "):", mean((test$y-predict(fit2,test))^2), "\n")

  # forward
  fit <- lm(y~1, train)
  fit2 <- step(fit, scope=~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10+X11+X12+X13+X14+X15,
    direction="both", trace=F)
  cat("forward (", fit2$rank-1, "):", mean((test$y-predict(fit2,test))^2), "\n")

  # you add code to fit ridge and lasso models here
  x_train = model.matrix(y~.-1, train)
  x_test = model.matrix(y~.-1, test)
  #Ridge
  fitridge <- cv.glmnet(x_train, train$y, alpha = 0)
  prediction <- predict(fitridge, s = fitridge$lambda.min, newx = x_test)
  cat("Ridge:", mean((test$y-prediction)^2), "\n")

  #Lasso
  fitlasso <- cv.glmnet(x_train, train$y)
  prediction <- predict(fitlasso, s = fitlasso$lambda.min, newx = x_test)
  cat("Lasso:", mean((test$y-prediction)^2), "\n")

  invisible(list(train=train, test=test))
}
```

3(n): We see that in the case of high multicollinearity, lasso usually performs the best. This is predictable, as it performs variable selection and shrinkage at the same time, which will be especially useful in this case where highly correlated independent variables will affect each other. Ridge also performs well, but not as good as lasso. Shrinkage generally performs well to reduce the contribution of variables x6-x15. Also stepwise performs worsen as the noise become lower, possibly due to small variance, and the AIC value will be affected sometimes more by the x6-x15 due to small noise, and they can happen to explain more variance in different combinations.

```r
hw5(rho = 0.9, sigmae = 5)
```

```
## -------------------------------------------
## correlation between x:  0.9
## Error variance:  25
##
## Call:
## lm(formula = y ~ X1 + X2 + X3 + X4 + X5, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.0727  -3.4071   0.4994   3.0555  11.5877
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0427     0.5491   5.541 2.73e-07 ***
## X1            0.9064     1.4791   0.613   0.5415
```

```
## X2           -2.0427     1.6748  -1.220   0.2256
## X3            3.6465     1.4873   2.452   0.0161 *
## X4           -0.8389     1.7398  -0.482   0.6308
## X5           -0.2851     1.3606  -0.210   0.8345
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.333 on 94 degrees of freedom
## Multiple R-squared:  0.1186, Adjusted R-squared:  0.07173
## F-statistic:  2.53 on 5 and 94 DF,  p-value: 0.03405
##
## OLS x1-x5: 26.24862
## OLS x1-x15: 28.4299
## backward ( 2 ): 26.96192
## forward ( 2 ): 26.96192
## Ridge: 26.08833
## Lasso: 25.70437
```

```r
hw5()
```

```
## -------------------------------------------
## correlation between x:  0.9
## Error variance:  9
##
## Call:
## lm(formula = y ~ X1 + X2 + X3 + X4 + X5, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.8436 -2.0442  0.2997  1.8333  6.9526
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0256     0.3295   9.183    1e-14 ***
## X1            0.9439     0.8875   1.064  0.29026
## X2           -1.6256     1.0049  -1.618  0.10906
## X3            2.7879     0.8924   3.124  0.00237 **
## X4           -0.3034     1.0439  -0.291  0.77200
## X5           -0.3711     0.8164  -0.455  0.65048
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.2 on 94 degrees of freedom
## Multiple R-squared:  0.2407, Adjusted R-squared:  0.2003
## F-statistic: 5.961 on 5 and 94 DF,  p-value: 7.843e-05
##
## OLS x1-x5: 9.449501
## OLS x1-x15: 10.23477
## backward ( 4 ): 10.04426
## forward ( 2 ): 9.879715
## Ridge: 9.570036
## Lasso: 9.41732
```

```r
hw5(rho = 0.9, sigmae = 1)
```

```
## -------------------------------------------
## correlation between x:  0.9
## Error variance:  1
##
## Call:
## lm(formula = y ~ X1 + X2 + X3 + X4 + X5, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.61454 -0.68141  0.09989  0.61110  2.31754
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0085     0.1098  27.395  < 2e-16 ***
## X1            0.9813     0.2958   3.317 0.001294 **
## X2           -1.2085     0.3350  -3.608 0.000497 ***
## X3            1.9293     0.2975   6.486 4.07e-09 ***
## X4            0.2322     0.3480   0.667 0.506172
## X5           -0.4570     0.2721  -1.680 0.096373 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.067 on 94 degrees of freedom
## Multiple R-squared:  0.7193, Adjusted R-squared:  0.7044
## F-statistic: 48.18 on 5 and 94 DF,  p-value: < 2.2e-16
##
## OLS x1-x5: 1.049945
## OLS x1-x15: 1.137196
## backward ( 4 ): 1.14975
## forward ( 4 ): 1.14975
## Ridge: 1.110243
## Lasso: 1.092613
```

For this case where we have moderate multicollinearity, shrinkage method still performs well, especially in high noise case. Shrinkage will also shrink the coefficient toward 0 if the coefficient is found unnecessary, which will be especially useful in high noise case. However, we see in the low noise case, stepwise(both backward and forward) performs well. Due to low noise and moderate covariance, the unaffect variable(x6-x15) will not display similar pattern with the true variable, so they will highly increase the AIC value possibly and being excluded.

```r
hw5(rho = 0.5, sigmae = 5)
```

```
## -------------------------------------------
## correlation between x:  0.5
## Error variance:  25
##
## Call:
## lm(formula = y ~ X1 + X2 + X3 + X4 + X5, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.0727  -3.4071   0.4994   3.0555  11.5877
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0427     0.5491   5.541 2.73e-07 ***
## X1            0.8773     0.6134   1.430 0.155995
## X2           -1.4325     0.7637  -1.876 0.063802 .
## X3            2.4677     0.6751   3.655 0.000423 ***
## X4           -0.1112     0.7949  -0.140 0.889065
## X5           -0.4022     0.6193  -0.649 0.517672
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.333 on 94 degrees of freedom
## Multiple R-squared:  0.1641, Adjusted R-squared:  0.1197
## F-statistic: 3.691 on 5 and 94 DF,  p-value: 0.004288
##
## OLS x1-x5: 26.24862
## OLS x1-x15: 28.4299
## backward ( 4 ): 27.28365
## forward ( 2 ): 26.96062
## Ridge: 27.1824
## Lasso: 26.84763
```

```r
hw5(rho = 0.5, sigmae = 3)
```

```
## ------------------------------------------
## correlation between x:  0.5
## Error variance:  9
##
## Call:
## lm(formula = y ~ X1 + X2 + X3 + X4 + X5, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.8436 -2.0442  0.2997  1.8333  6.9526
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0256     0.3295   9.183 1.00e-14 ***
## X1            0.9264     0.3681   2.517  0.01353 *
## X2           -1.2595     0.4582  -2.749  0.00718 **
## X3            2.0806     0.4051   5.136 1.51e-06 ***
## X4            0.1333     0.4769   0.279  0.78050
## X5           -0.4413     0.3716  -1.188  0.23799
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.2 on 94 degrees of freedom
## Multiple R-squared:  0.3092, Adjusted R-squared:  0.2724
## F-statistic: 8.414 on 5 and 94 DF,  p-value: 1.31e-06
##
## OLS x1-x5: 9.449501
## OLS x1-x15: 10.23477
## backward ( 4 ): 10.06145
## forward ( 4 ): 10.06145
## Ridge: 9.724438
```

```
## Lasso: 9.712083
```

```r
hw5(rho = 0.5, sigmae = 1)
```

```
## ---------------------------------------------
## correlation between x:  0.5
## Error variance:  1
##
## Call:
## lm(formula = y ~ X1 + X2 + X3 + X4 + X5, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.61454 -0.68141  0.09989  0.61110  2.31754
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0085     0.1098  27.395  < 2e-16 ***
## X1            0.9755     0.1227   7.951 4.05e-12 ***
## X2           -1.0865     0.1527  -7.113 2.21e-10 ***
## X3            1.6935     0.1350  12.542  < 2e-16 ***
## X4            0.3778     0.1590   2.376 0.019524 *
## X5           -0.4804     0.1239  -3.879 0.000195 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.067 on 94 degrees of freedom
## Multiple R-squared:  0.771,  Adjusted R-squared:  0.7588
## F-statistic: 63.31 on 5 and 94 DF,  p-value: < 2.2e-16
##
## OLS x1-x5: 1.049945
## OLS x1-x15: 1.137196
## backward ( 6 ): 1.093831
## forward ( 6 ): 1.093831
## Ridge: 1.104976
## Lasso: 1.101268
```

We saw similar pattern with moderate multicollinearity here, Lasso still performs really well in every case. Ridge regression however performs poorly in high and low noise case. As expected, stepwise selection performs better and better as the noice get smaller, and forward and backward give the same results

```r
hw5(rho = 0.1, sigmae = 5)
```

```
## ---------------------------------------------
## correlation between x:  0.1
## Error variance:  25
##
## Call:
## lm(formula = y ~ X1 + X2 + X3 + X4 + X5, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.0727  -3.4071   0.4994   3.0555  11.5877
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  3.04265    0.54911   5.541 2.73e-07 ***
## X1           0.88334    0.48641   1.816   0.0726 .
## X2          -1.27395    0.59604  -2.137   0.0352 *
## X3           2.22519    0.53174   4.185 6.42e-05 ***
## X4           0.02041    0.62941   0.032   0.9742
## X5          -0.42305    0.48730  -0.868   0.3875
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.333 on 94 degrees of freedom
## Multiple R-squared:  0.1973, Adjusted R-squared:  0.1546
## F-statistic: 4.621 on 5 and 94 DF,  p-value: 0.0008171
##
## OLS x1-x5: 26.24862
## OLS x1-x15: 28.4299
## backward ( 4 ): 27.49461
## forward ( 4 ): 27.49461
## Ridge: 27.86595
## Lasso: 26.71499
```

```r
hw5(rho = 0.1, sigmae = 3)
```

```
## ------------------------------------------
## correlation between x:  0.1
## Error variance:  9
##
## Call:
## lm(formula = y ~ X1 + X2 + X3 + X4 + X5, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.8436 -2.0442  0.2997  1.8333  6.9526
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.0256     0.3295   9.183 1.00e-14 ***
## X1            0.9300     0.2918   3.187  0.00195 **
## X2           -1.1644     0.3576  -3.256  0.00157 **
## X3            1.9351     0.3190   6.065 2.74e-08 ***
## X4            0.2122     0.3776   0.562  0.57544
## X5           -0.4538     0.2924  -1.552  0.12398
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.2 on 94 degrees of freedom
## Multiple R-squared:  0.3608, Adjusted R-squared:  0.3268
## F-statistic: 10.61 on 5 and 94 DF,  p-value: 4.24e-08
##
## OLS x1-x5: 9.449501
## OLS x1-x15: 10.23477
## backward ( 4 ): 10.27753
## forward ( 4 ): 10.27753
## Ridge: 9.941478
## Lasso: 9.840797
```

```r
hw5(rho = 0.1, sigmae = 1)
```

```
## -------------------------------------------
## correlation between x:  0.1
## Error variance:  1
##
## Call:
## lm(formula = y ~ X1 + X2 + X3 + X4 + X5, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.61454 -0.68141  0.09989  0.61110  2.31754
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.00853    0.10982  27.395  < 2e-16 ***
## X1           0.97667    0.09728  10.040  < 2e-16 ***
## X2          -1.05479    0.11921  -8.848 5.16e-14 ***
## X3           1.64504    0.10635  15.468  < 2e-16 ***
## X4           0.40408    0.12588   3.210  0.00182 **
## X5          -0.48461    0.09746  -4.972 2.97e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.067 on 94 degrees of freedom
## Multiple R-squared:  0.8089, Adjusted R-squared:  0.7988
## F-statistic:  79.6 on 5 and 94 DF,  p-value: < 2.2e-16
##
## OLS x1-x5: 1.049945
## OLS x1-x15: 1.137196
## backward ( 6 ): 1.095551
## forward ( 6 ): 1.095551
## Ridge: 1.121333
## Lasso: 1.11112
```