Brief report

Outliers:

1. Summary of the data
2. Best set, forward, backward
3. Lasso (not use ridge because it does not do variable selection)
4. Compare models
5. Summary of the data

> names(Data)

[1] "Cdc42" "Pla2g6" "Akt2" "Plcg2" "Mapk1" "Rac2" "Rik"

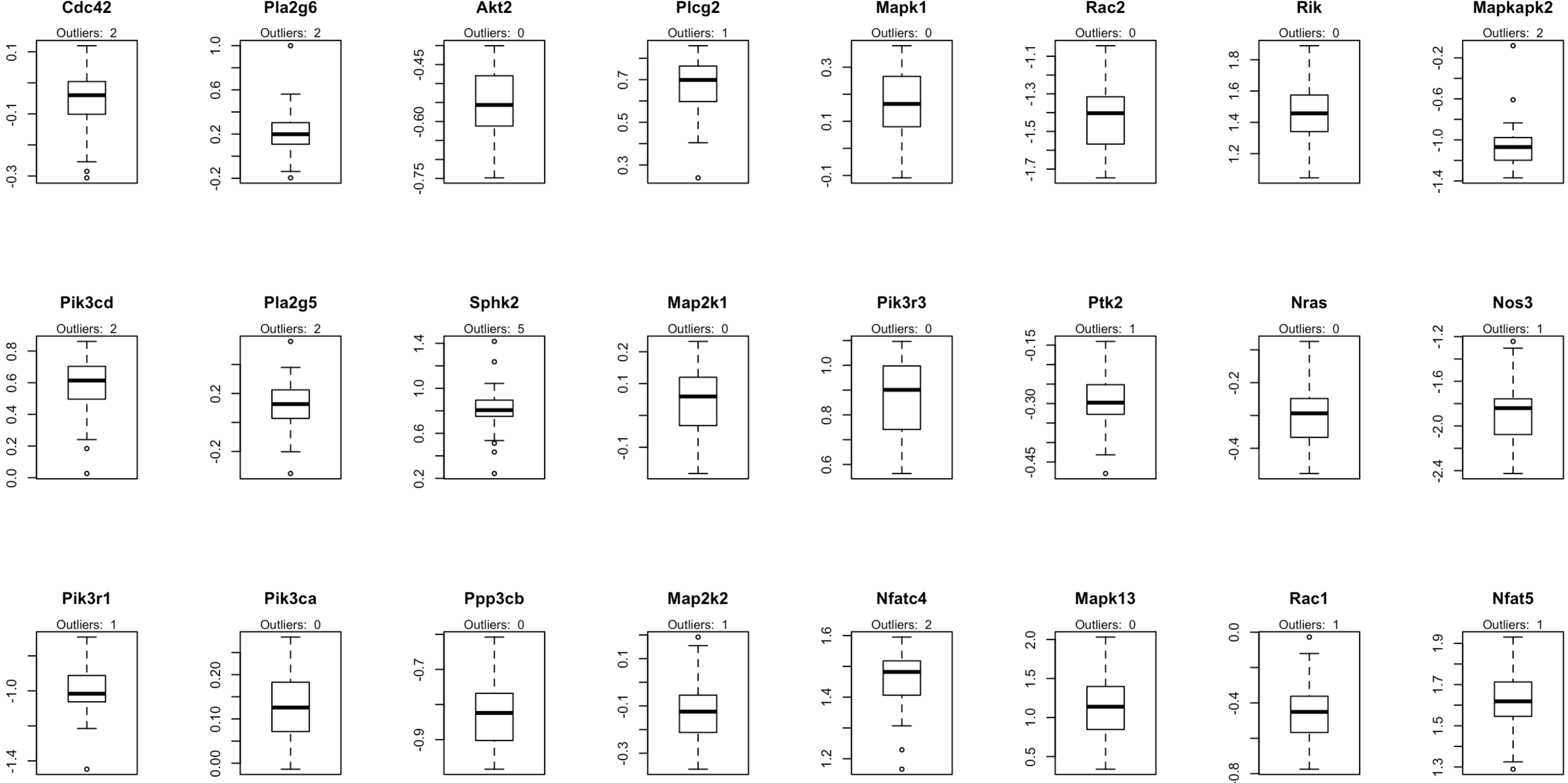
[8] "Mapkapk2" "Pik3cd" "Pla2g5" "Sphk2" "Map2k1" "Pik3r3" "Ptk2"

[15] "Nras" "Nos3" "Pik3r1" "Pik3ca" "Ppp3cb" "Map2k2" "Nfatc4"

[22] "Mapk13" "Rac1" "Nfat5"

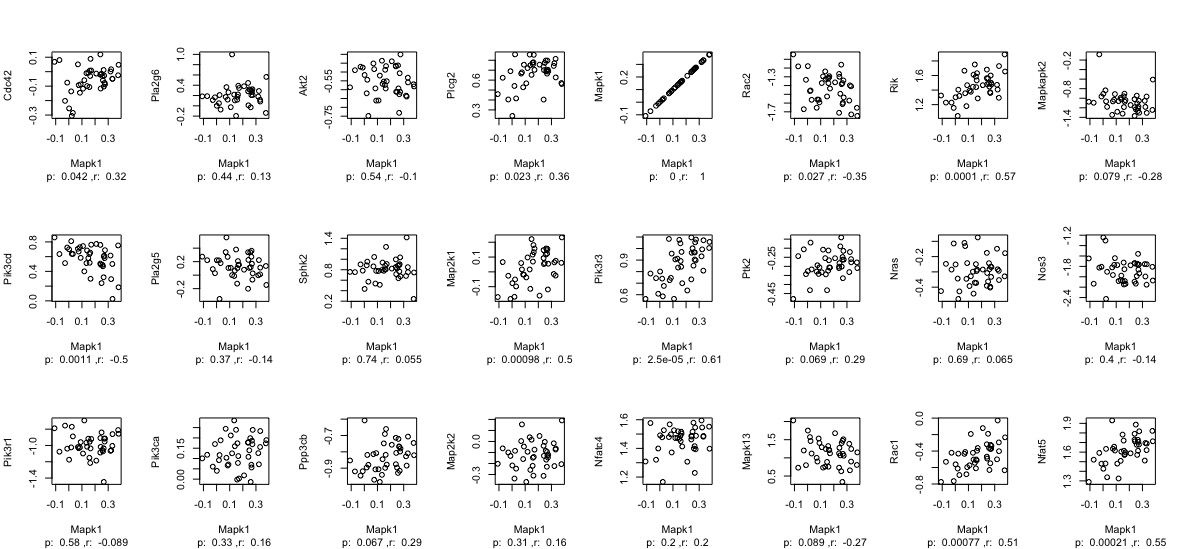
For the sake of simplicity, they are assigned number #1-#24 to indicate them

Check the distribution of all the data:



There are some outliers in some variables (based on 1.5 IQR). In the later analysis I did not exclude them, because the sample size is small (n = 40); if all the outliers are excluded, the sample pool will be reduced to about 25.

Check correlation between Mapk1 and other variables:



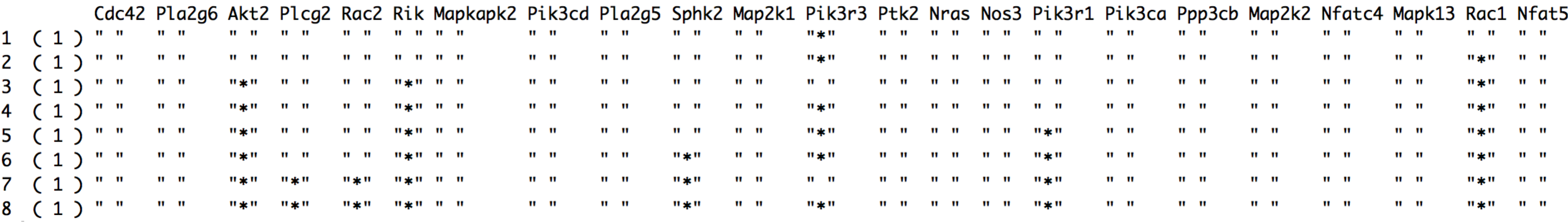
Significant correlation (possible predictors): 1\*, 4\*, 6\*, 7\*\*\*, 9\*\*, 12\*\*\*, 13\*\*\*, 23\*\*\*, 24\*\*\*

\* p < 0.05; \*\* p < 0.01; \*\*\* p < 0.001

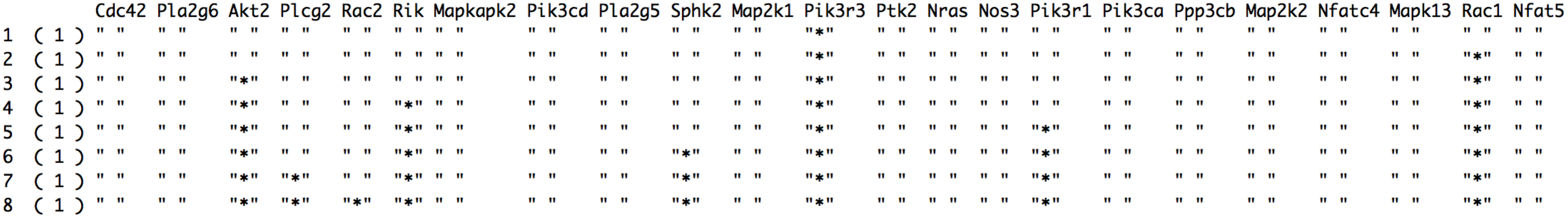
1. Best set, forward and backward models

* Results of the three approaches (in red squares are variable sets selected by Cp and BIC)

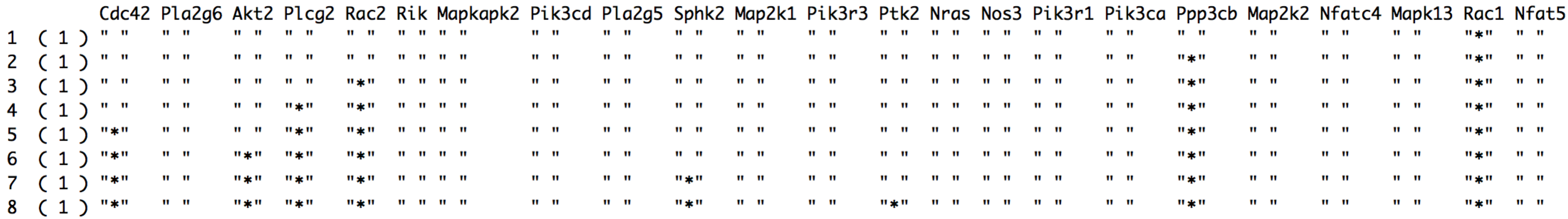
1. Best set:



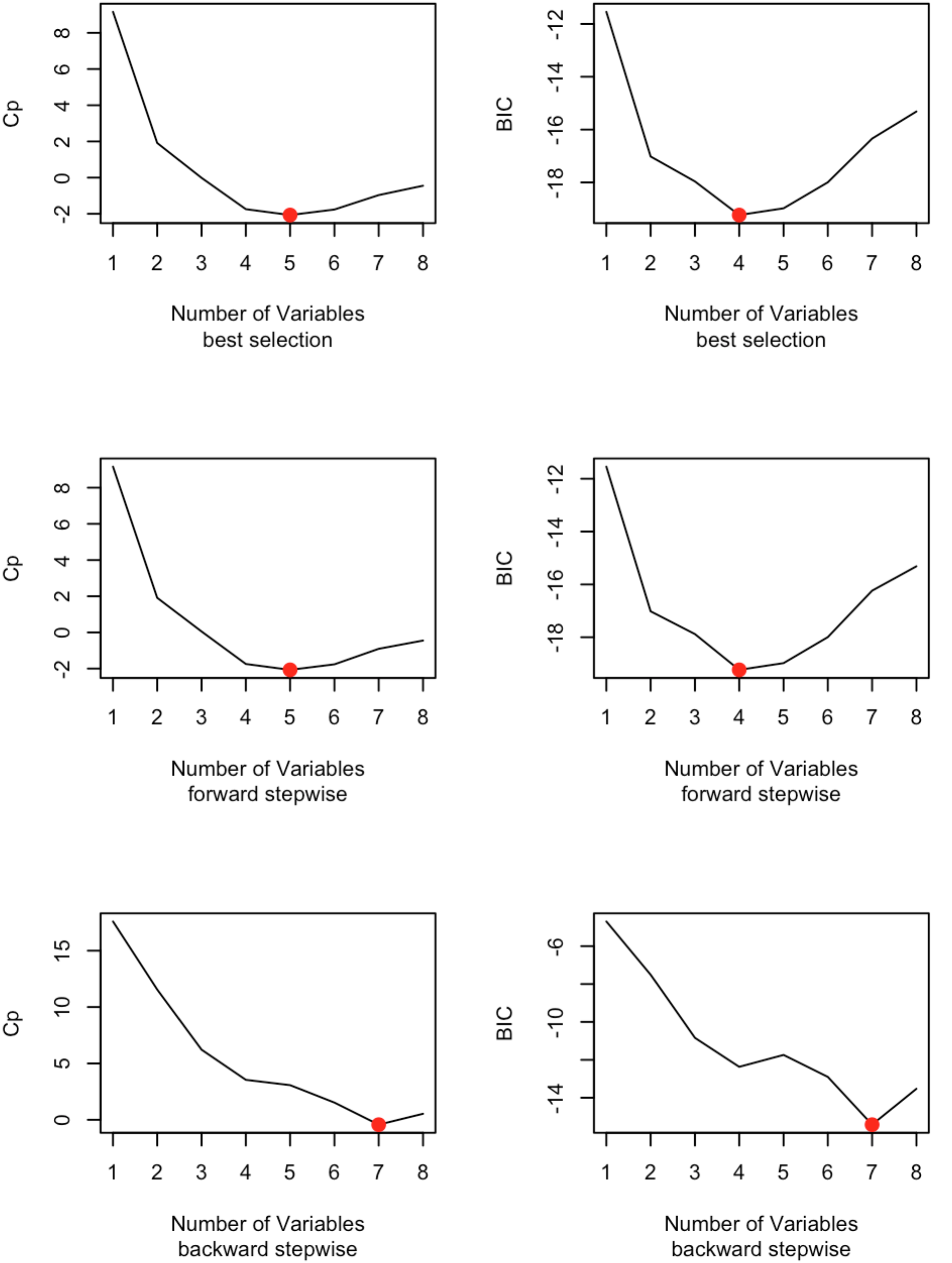
1. Forward (very similar to the set selected by best set):



1. Backward (quite different from the results of the best set and forward stepwise):



* Evaluation of the three models using Cp and BIC



* Best subset and forward method:

The best model selected by Cp: #3, #7, #13, #17, #23

(Intercept) Akt2 Rik Pik3r3 Pik3r1 Rac1

-0.6184661 -0.3965544 0.2496411 0.2235856 -0.1619541 0.3302738

The best model selected by BIC: #3, #7, #13, #23

(Intercept) Akt2 Rik Pik3r3 Rac1

-0.4446112 -0.4054803 0.2207164 0.2439920 0.3171645

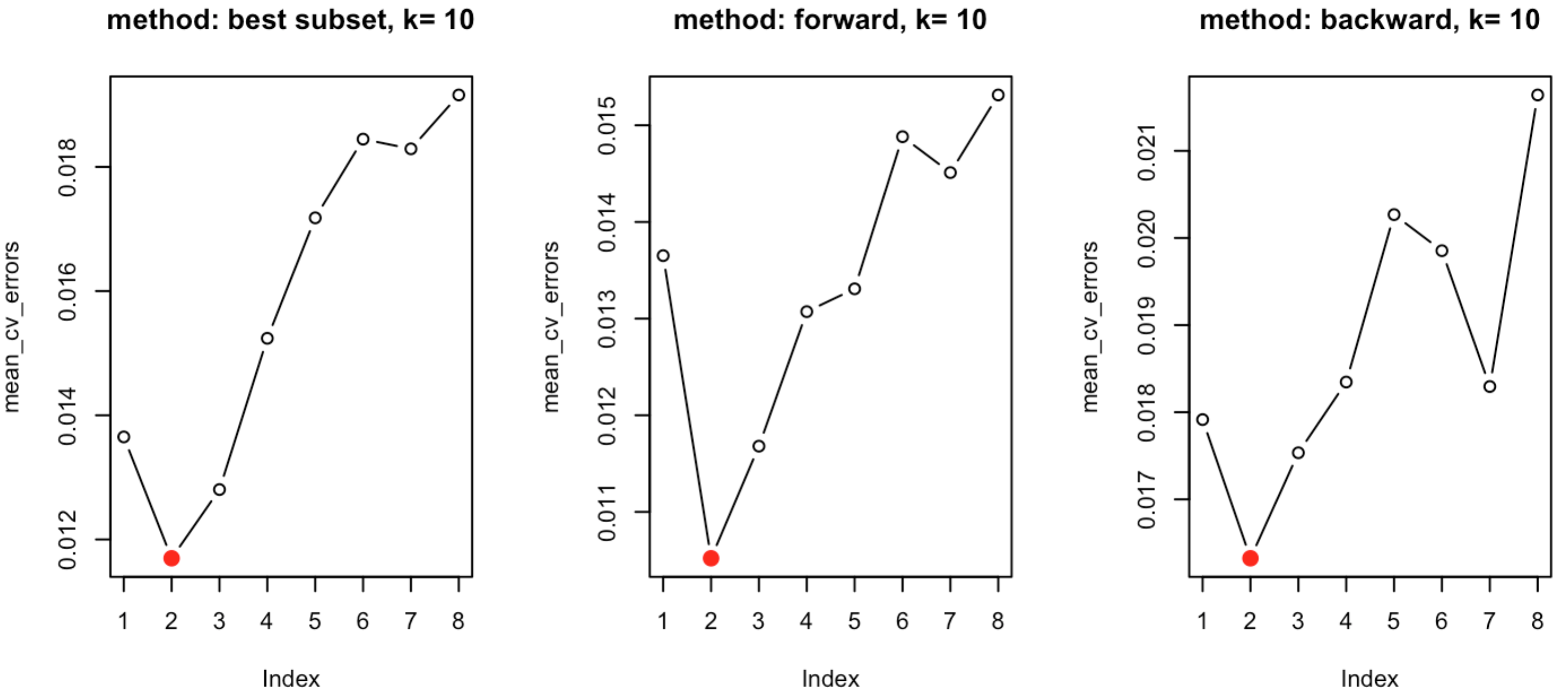
* Backward method:

Both Cp and BIC select #1, #3, #4, #6, #11, #19, #23

(Intercept) Cdc42 Akt2 Plcg2 Rac2 Sphk2 Ppp3cb Rac1

0.07666044 0.37749256 -0.38665341 0.28977515 -0.25753671 -0.17559530 0.43222257 0.35938288

Use 10-fold cross-validation[[1]](#footnote-1)



Best subset and foward: (#13, #23)

(Intercept) Pik3r3 Rac1

-0.05642834 0.40410424 0.27832221

Backward: (#19, #23)

(Intercept) Ppp3cb Rac1

0.7430295 0.4716582 0.4022791

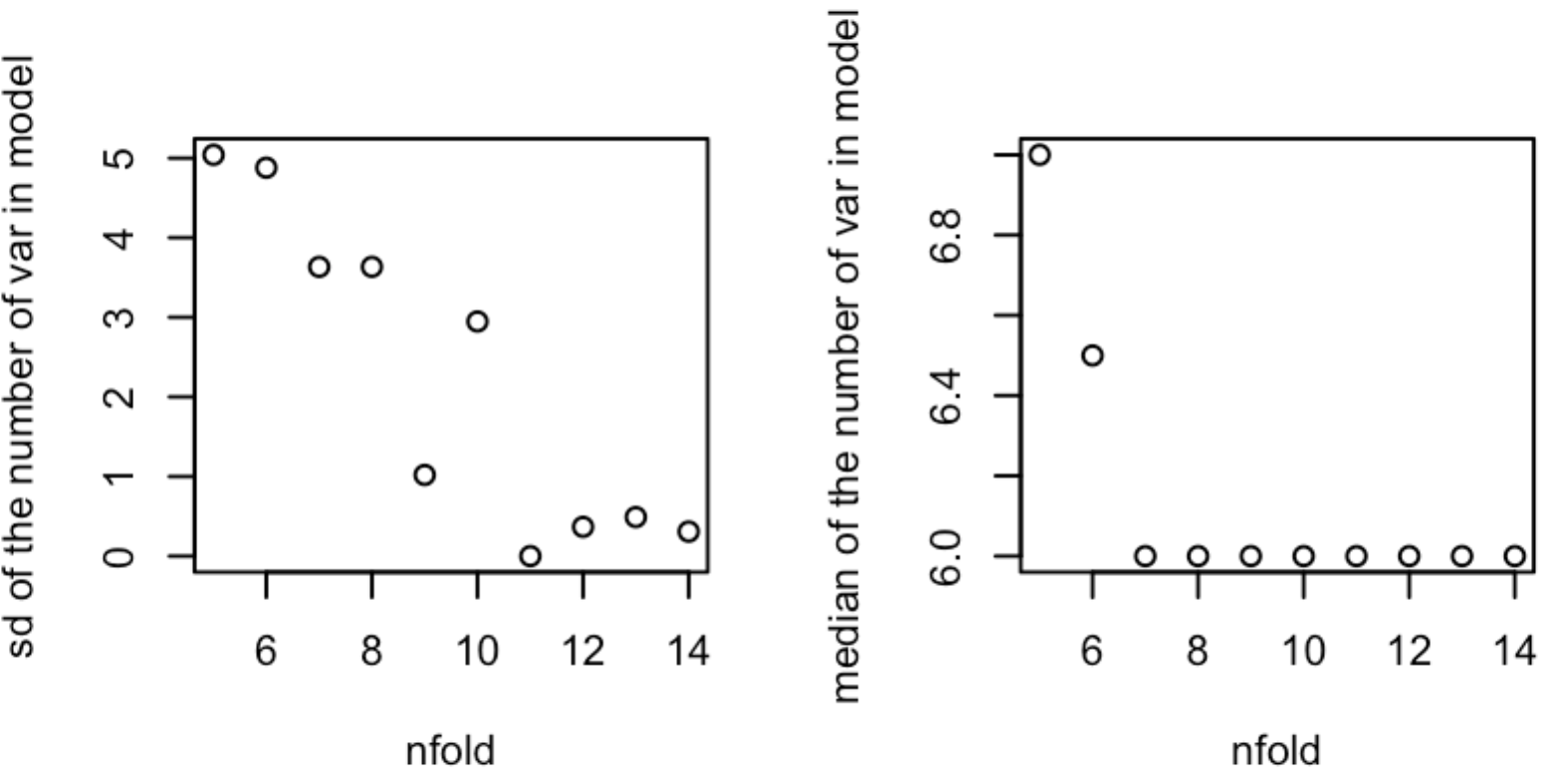
Compared to using Cp and BIC, cross-validation approach yields significantly smaller models.

1. Lasso: #7, #9, #13, #23, #24

I used cross-validation to obtain the best lambda that gives the lowest test MSE, and then used the best lambda to obtain the coefficients that are not zero. This gave the final model with a small number of selected variables.

The final model changes depending on the grid used[[2]](#footnote-2) and the number of folds in the cross validation, and varies when set.seed is set different (variation).

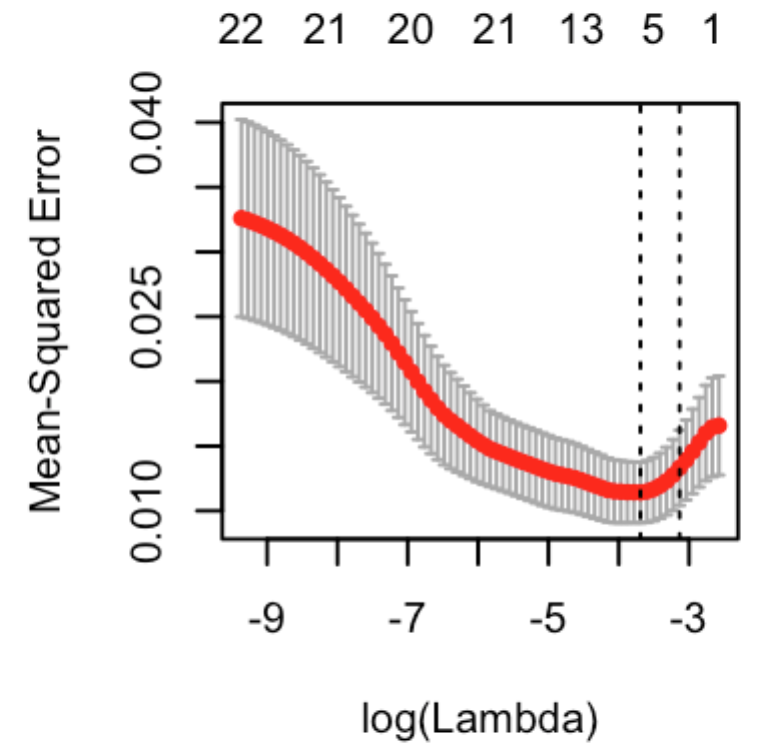
To obtain a stable model using this method, I tested how the number of CV folds (nfold) can influence the selected variables in the final model. I used the number of selected variable (*n*) as a simple indicator of whether the final model is the same or different. I obtained the standard deviation and median of *n* by setting set.seed different (so every time the training/test set are different). The results are present below:



As nfold increases, sd decreases (less variable); also the median is stable at 6 from nfold = 7. Therefore when nfold is too small, the stable results given by this method should include 6 variables, which are listed below: #7, #9, #13, #23, #24

(Intercept) Rik Pik3cd Pik3r3 Rac1 Nfat5

-0.13205963 0.10555116 -0.03199517 0.16367227 0.13200524 0.05115234



The lambda is 0.02493913

1. Compare models:

The above methods gave 6 models:

(1). #13, #23 (Best/Forward CV)

(2). #19, #23 (Backward CV)

(3). #3, #7, #13, #23 (Best/Forward BIC)

(4). #3, #7, #13, #17, #23 (Best/Forward Cp)

(5). #7, #9, #13, #23, #24 (Lasso)

(6). #1, #3, #4, #6, #11, #19, #23 (Backward BIC/Cp)

1. I did not use validation approach set because the sample size is small, the training and test set would be 20 if the validation set approach were used, which would yield results (the number of variables) with large variation (tested by using different set.seed(n)). The number of folds is set to be 10 instead of 5, because when k=5, the results also have larger variation, while the variation is very small when k = 10: always 2 predictors for forward and best subset method, and around 2 for backward method (larger variation than the other two). [↑](#footnote-ref-1)
2. I used the default grid values given in glmnet/cv.glmnet. I also tried varying n1, n2 and l in grid=10^seq(n1,n2,length= l): when n1 is no less than -1 but not very big, n2 is no bigger than -1 but not very small, and l is big enough, the grid is fine-grained enough to yield stable results. One example set of these values are n1 = 0, n2 = -4 and l = 200. Using these values gave the same results as using the default grid values. Therefore I do not go into details of how different grid can influence the results. [↑](#footnote-ref-2)