

# Comparison of mBIC2 information criterion with methods provided in L0Learn library

Aleksander Milach

22/02/2021

We provide results of the simulation study comparing mBIC2 information criterion with functions from *L0Learn* library. In all simulations we consider a multiple regression model with  $n = p = 500$ , the error term  $\epsilon$  is iid normally distributed  $N(0, I)$ , the number  $k$  of nonzero coefficients in the vector  $\beta$  takes values  $k \in \{10, 20, 40, 60, 80, 100\}$ . We distinguish two levels of signal strenght. We associate weak signal with

$$\beta_1 = \beta_2 = \dots = \beta_k = 1.3\sqrt{2\log p}$$

and strong signal with

$$\beta_1 = \beta_2 = \dots = \beta_k = 2\sqrt{2\log p}.$$

The rows of design matrix  $X$  are iid random vectors from multivariate normal distribution  $N\left(0, \frac{1}{n}\Sigma\right)$ .

We consider two scenarios, one with independent regressors, where  $\Sigma = I$ , and one with correlated regressors, where  $\Sigma_{i,j} = 1$ , when  $i = j$  and  $\Sigma_{i,j} = 0.5$ , when  $i \neq j$ .

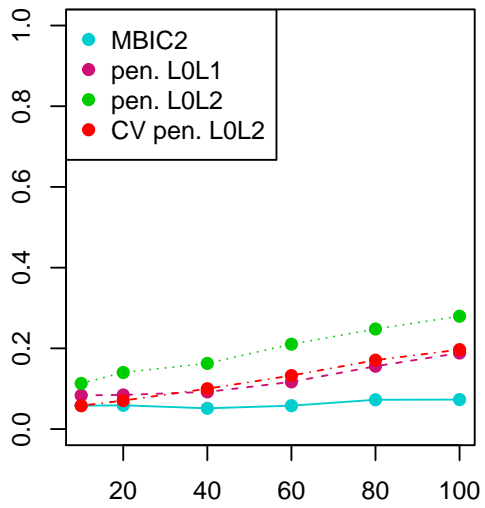
For each simulation we calculate values of four statistics: false discovery rate (**FDR**) defined as average of  $\#FP/\max(1, \#TP + \#FP)$ , **power** defined as  $\#TP/k$ , **MSE** =  $MSE(\hat{\beta}) / \|\beta\|^2$  and **MSP** =  $MSE(X\hat{\beta}) / \|X\beta\|^2$ . The results are based on 200 simulation runs.

The first simulation compares the following methods:

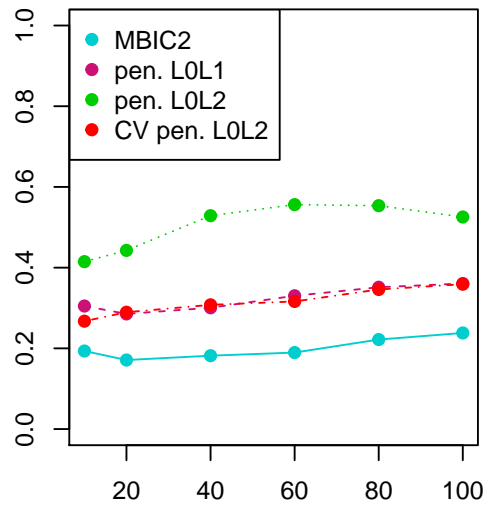
- **mBIC2** with stepwise searching procedure provided in *bigstep* library (cyan curve),
- *L0Learn.fit* with **penalty L0L1** with the most accurate support size available (purple curve),
- *L0Learn.fit* with **penalty L0L2** with the most accurate support size available (green curve),
- *L0Learn.fit* with **CV penalty L0L2** - cross validation is performed firstly on  $\gamma$  and then on  $\lambda$  (red curve).

By selecting the most accurate support size available we mean picking from object generated by *L0Learn.fit* the model, which support size is closest to  $k$ . Such rule is very theoretical and requires a very strong assumption, but it is easy to come up with to start off and get first results. The idea of this experiment was to try to use functions from *L0Learn* library, so the results should not be examined too carefully, notably the fourth experiment is precisely the example usage of *L0Learn.cvfit* from *L0Learn* library vignette.

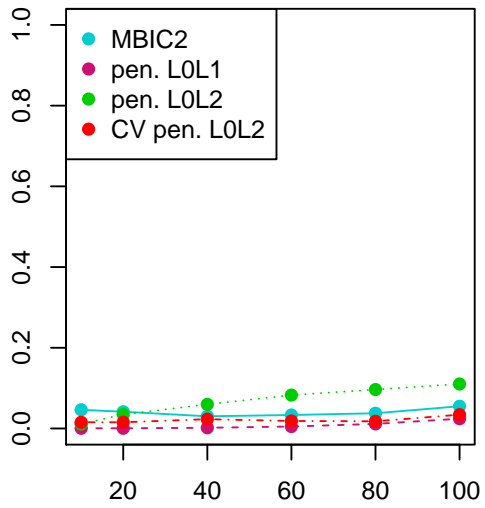
**FDR\_NoCorr\_Weak**



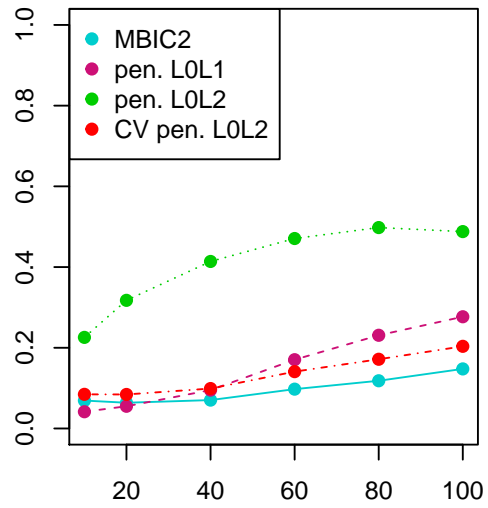
**FDR\_Corr\_Weak**

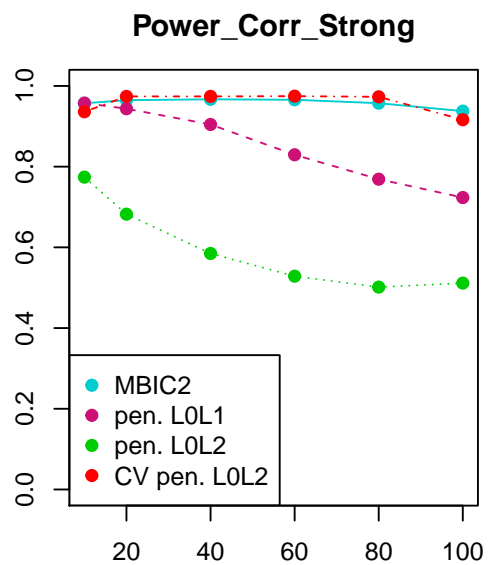
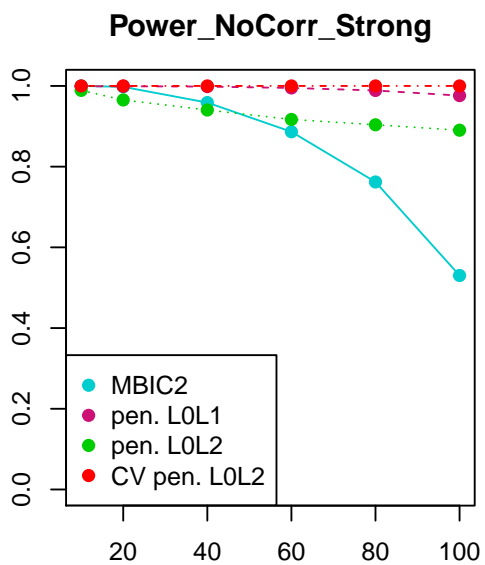
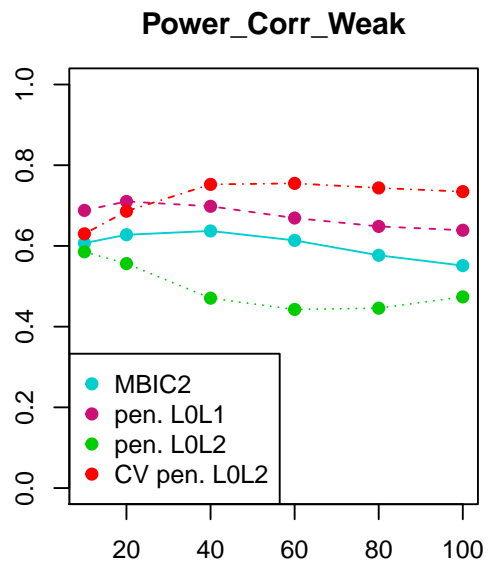
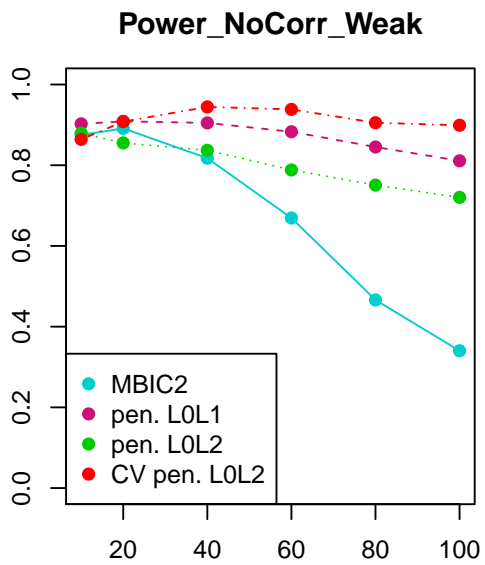


**FDR\_NoCorr\_Strong**

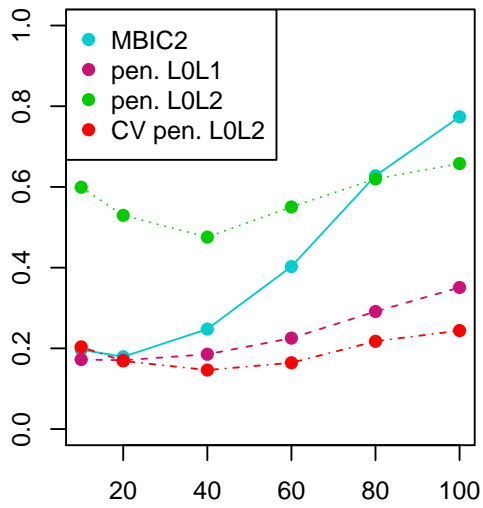


**FDR\_Corr\_Strong**

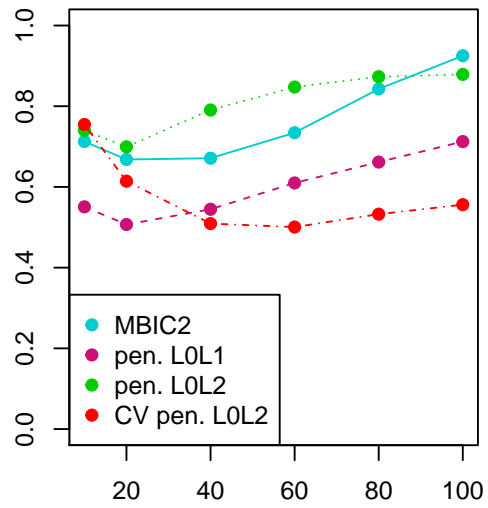




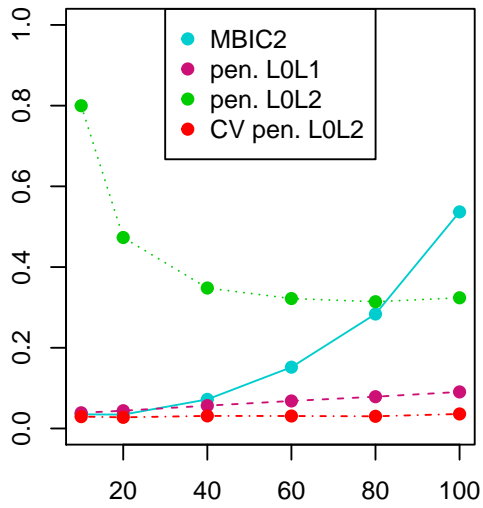
**MSE\_NoCorr\_Weak**



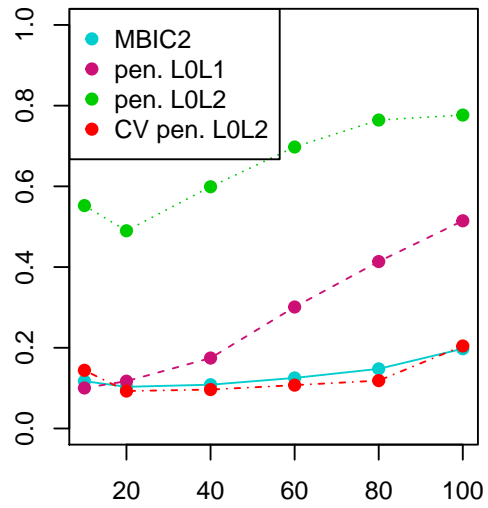
**MSE\_Corr\_Weak**

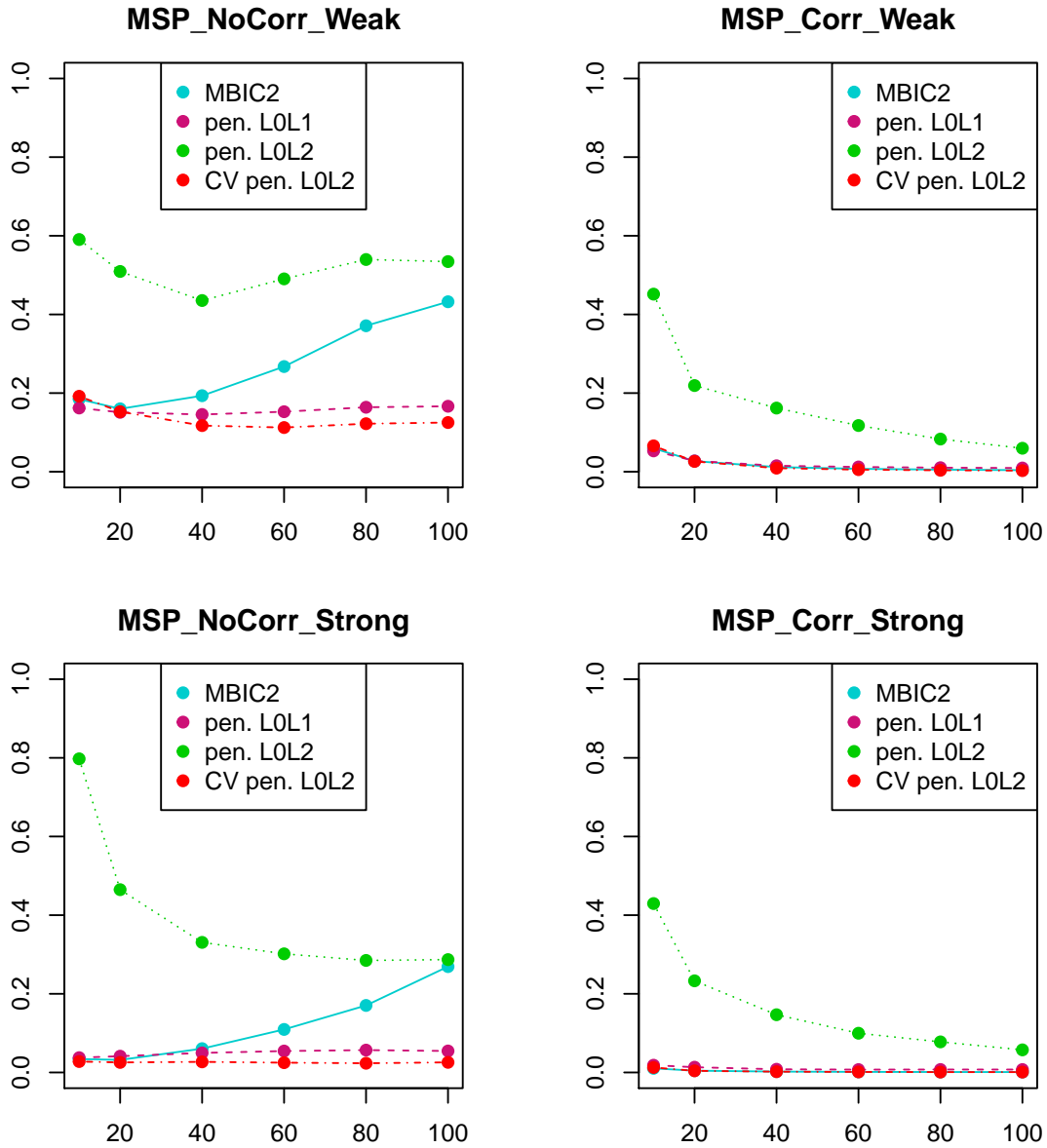


**MSE\_NoCorr\_Strong**



**MSE\_Corr\_Strong**

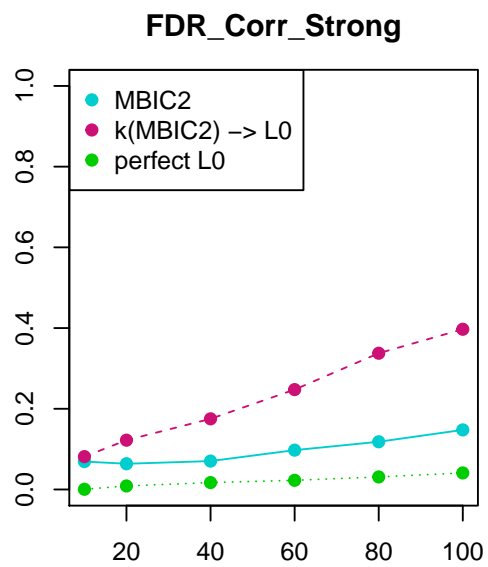
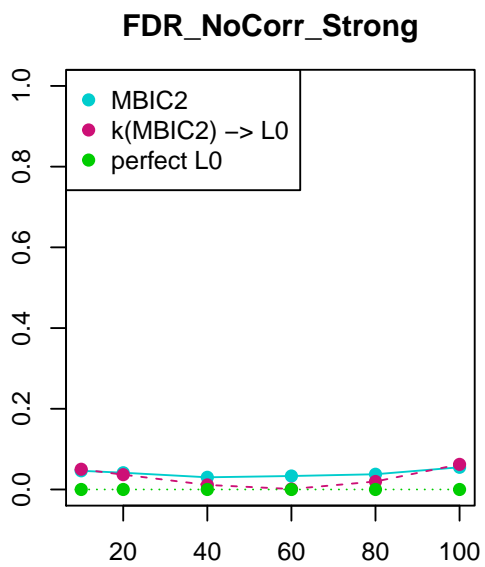
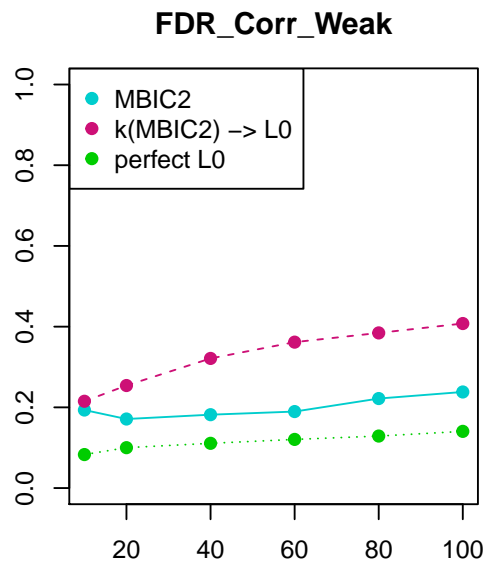
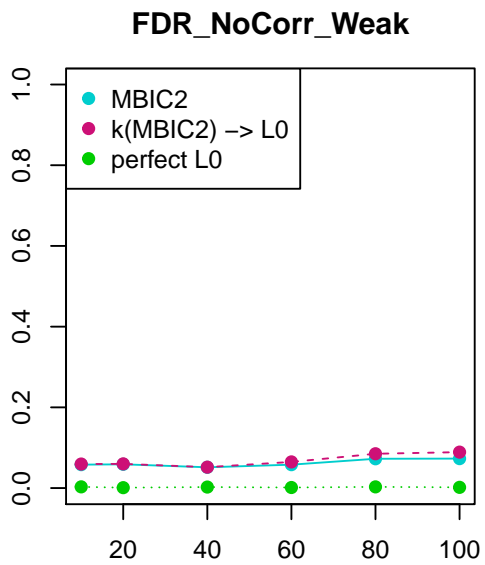


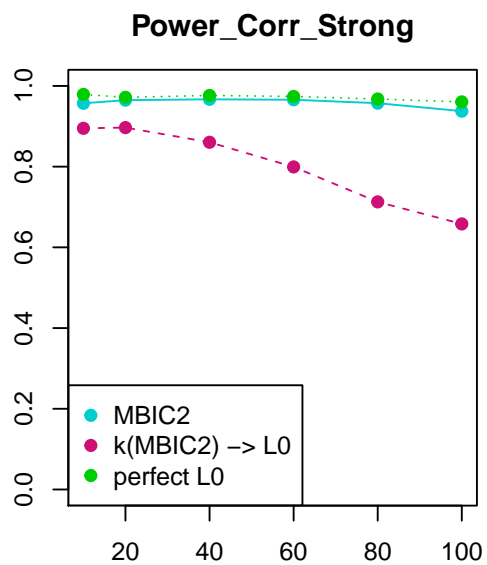
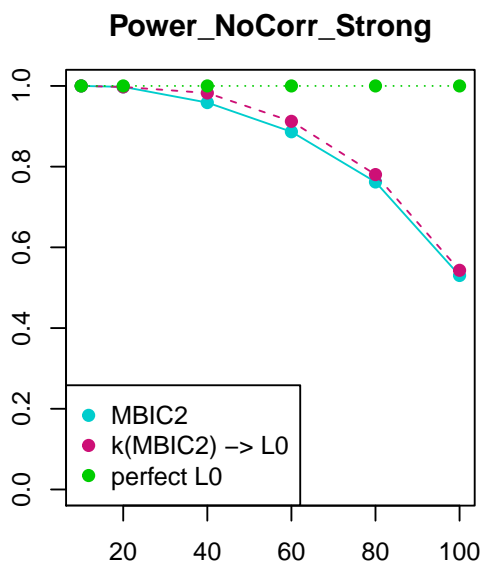
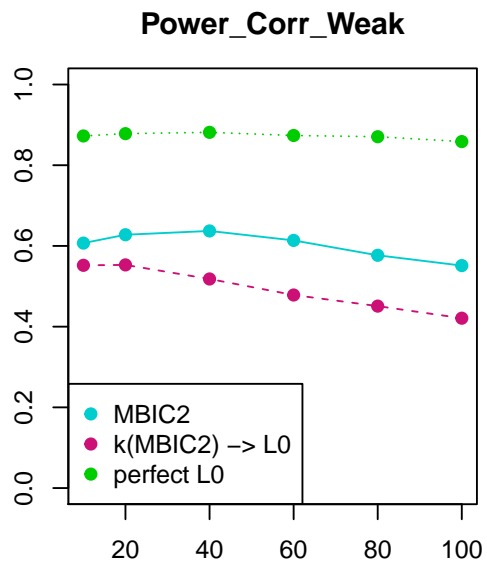
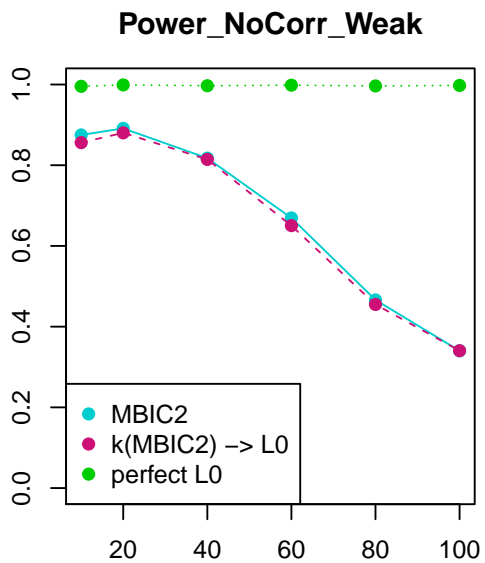


The second simulation compares the following methods:

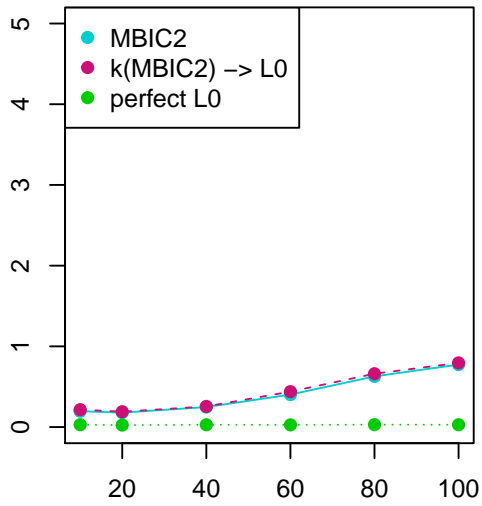
- **mBIC2** with stepwise searching procedure provided in *bigstep* library - same as in the first simulation (cyan curve)
- *L0Learn.fit* with **penalty L0** with the most accurate support size available (green curve),
- *L0Learn.fit* **penalty L0** with the support size most accurate to mBIC2 (purple curve).

Here we focus only on the optimisation with only L0 penalty. The second experiment is similar to two from in the first simulation. The third experiment is an attempt to apply a more serious method of selecting model from object returned by *L0Learn.fit*. On the contrary to the first approach, we choose the model, which support size is closest to the one chosen by mBIC2.

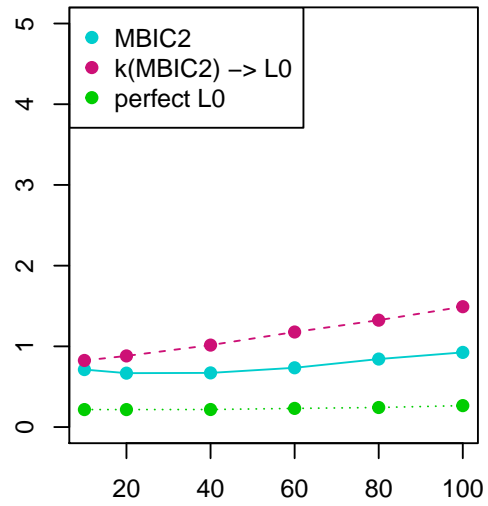




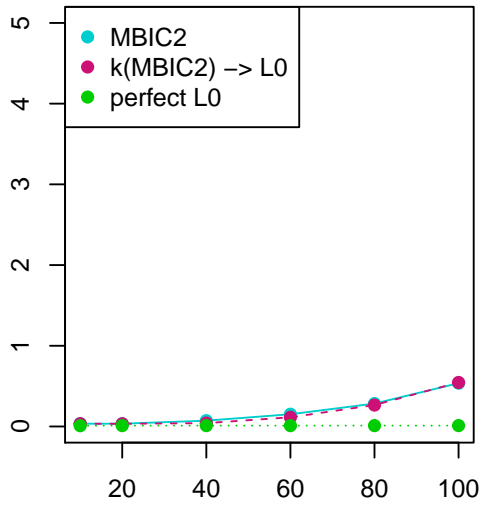
**MSE\_NoCorr\_Weak**



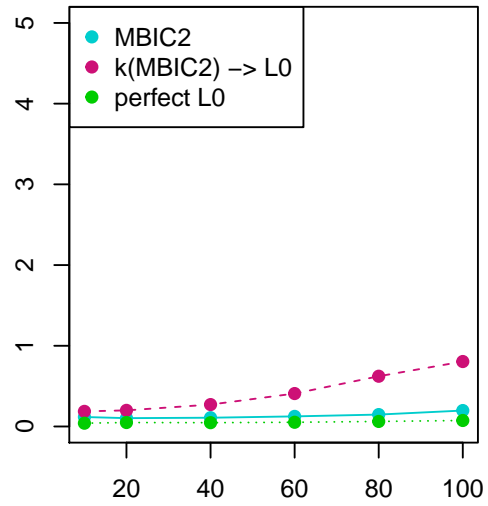
**MSE\_Corr\_Weak**



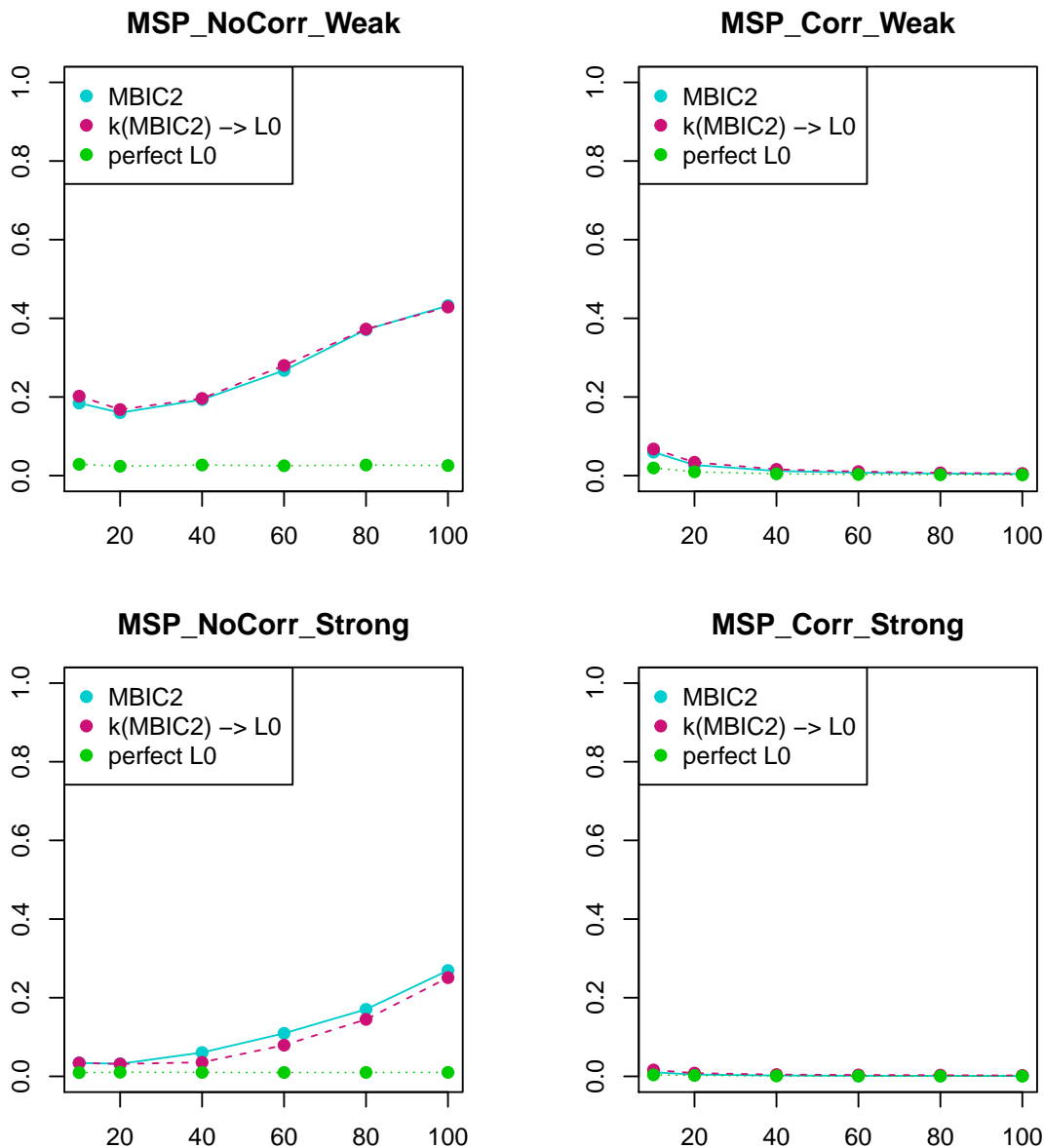
**MSE\_NoCorr\_Strong**



**MSE\_Corr\_Strong**







Here we focus only on the optimisation with only L0 penalty. The second experiment is similar to two from in the first simulation. The third experiment is an attempt to apply a more serious method of selecting a model from object returned by *L0Learn.fit*. In the contrary to the first approach, we choose the model, which support size is closest to the support size of the one chosen by mBIC2.

We can notice, that if we can correctly estimate the number of nonzero coefficients, we obtain very good results. The output of *L0Learn.fit* very frequently contains a good model, so discovering a well-performing method of selection should be further investigated. Thus we can associate the green curve with the target results of further research.

We see, that when the regressors are independent, the results of this method and mBIC2 are almost equal, with some small differences in FDR and power in the favor of *L0Learn.fit*. In case of correlation mBIC2 produces better results, with the differences being larger in strong signal case.