SVD Dimension reduction method

Xuelong Wang

2018-11-19

Contents

1 Motivation **1**

2 Main idea two steps **1**
2.1 Dimension Reduction . . . 1
2.2 Following with GCTA method . . . 2

3 Simulation study **2**
3.1 Simulation setup . . . 2

4 PCBs data simulation result **6**
4.1 sample matrix of PCB data . . . 6
4.2 Simulation result . . . 8

1 Motivation

Based on previous simulation results we did a series of simulation on estimation of total variance of main and interactive effects. we found that combing dimension reduction with decorrelation tend (our proposed method) to have a better result than GCTA, especailly when n < p and correlation between covariates are high. Therefore, we condcuted a group of simulation studies trying to evaluate the performance of the proposed method. we tried different covariance structures and PCBs data with re-sampling. Overall, the performance is good in most of the case. When n is small and correlation is also weak, the prospoed method is as good as the original GCTA method.

2 Main idea two steps

2.1 Dimension Reduction

$$
\begin{aligned}
X = UDV^T &= \begin{bmatrix} U_r & U_2 \end{bmatrix} \begin{bmatrix} D_r & 0 \\ 0 & D_2 \end{bmatrix} \begin{bmatrix} V_r & V_2 \\ V_3 & V_4 \end{bmatrix}^T \\
&= \begin{bmatrix} U_r D_r & U_2 D_2 \end{bmatrix} \begin{bmatrix} V_r^T & V_3^T \\ V_2^T & V_4^T \end{bmatrix} = \begin{bmatrix} U_r D_r V_r^T + U_2 D_2 V_2^T & U_r D_r V_3^T + U_2 D_2 V_4^T \end{bmatrix}
\end{aligned}
$$

Ignore V_2, V_3 and V_4 , then we have the X_r as following

$$X_r = U_r D_r V_r^T.$$

We use X_r as the new covariates to the proposed methd. Therefore, we reduce the dimension from p to n

2.2 Following with GCTA method

After calculating X_r, we can regard X_r as our new predictors and use it as the input to the proposed method Note that we could use this blocking method to reduce X's dimension to $k, k \leq min(p, n)$.

3 Simulation study

I used Chi-square random variable with df = 1. To generate a certain covariance structure, one could randomly generate a sample from multivariate-normal-distribution first, and then just square each elements to have a group univarate Chi-saure distribution with desired correlations. The details of simulation is shown as follows.

3.1 Simulation setup

1. Normal distribution

$$X = [X_1 \ldots, X_p] \quad cov(X_i, X_j) = \Sigma_X$$

2. Chi-square distribution

$$T = [T_1 \ldots, T_p], \quad T_i = X_i^2 \sim \chi^2_{(1)}, \quad cov(T_i, T_j) = \Sigma_{\chi^2}$$

- The sample size n is from 100 to 800
- The number of main effect is 34 (p = 34)

3.1.1 correlation of T_i and T_j

Assume $Cov(X_i, X_j) = \sigma_{ij}, \quad Var(X_i) = \sigma_i^2$, $E(X_i) = 0$ and constant variance, then we have

$$Var(X_i) = E(X_i^2) - E(X_i)^2 = E(X_i^2) = \sigma_i^2 = \sigma^2$$

$$\begin{aligned} Cov(T_i, T_j) = Cov(X_i^2, X_j^2) &= E\left((X_i^2 - E(X_i^2))(X_j^2 - E(X_j^2))\right) \\ &= E(X_i^2 X_j^2 - X_i^2 E(X_j^2) - X_j^2 E(X_i^2) + E(X_i^2)E(X_j^2)) \\ &= E(X_i^2 X_j^2) - \sigma^4 \\ &= \sigma_i^2 \sigma_j^2 + 2\sigma_{ij}^2 - \sigma^4 \\ &= 2\sigma_{ij}^2 \end{aligned}$$

$$\begin{aligned} Cor(T_i, T_j) &= \frac{Cov(X_i^2, X_j^2)}{\sqrt{Var(X_i^2)Var(X_j^2)}} \\ &= \frac{2\sigma_{ij}^2}{2\sigma^4} \\ &= \frac{2(\rho\sigma^2)^2}{2\sigma^4} \\ &= \rho^2 \end{aligned}$$

3.1.2 Compound Symmetry

$$T = [T_1 \ldots, T_p], \quad T_i \sim \chi^2_{(1)}, \quad cov(T_i, T_j) = 2\rho^2, \quad \forall i \neq j, \rho = \{0.1, \ldots, 0.9\}$$

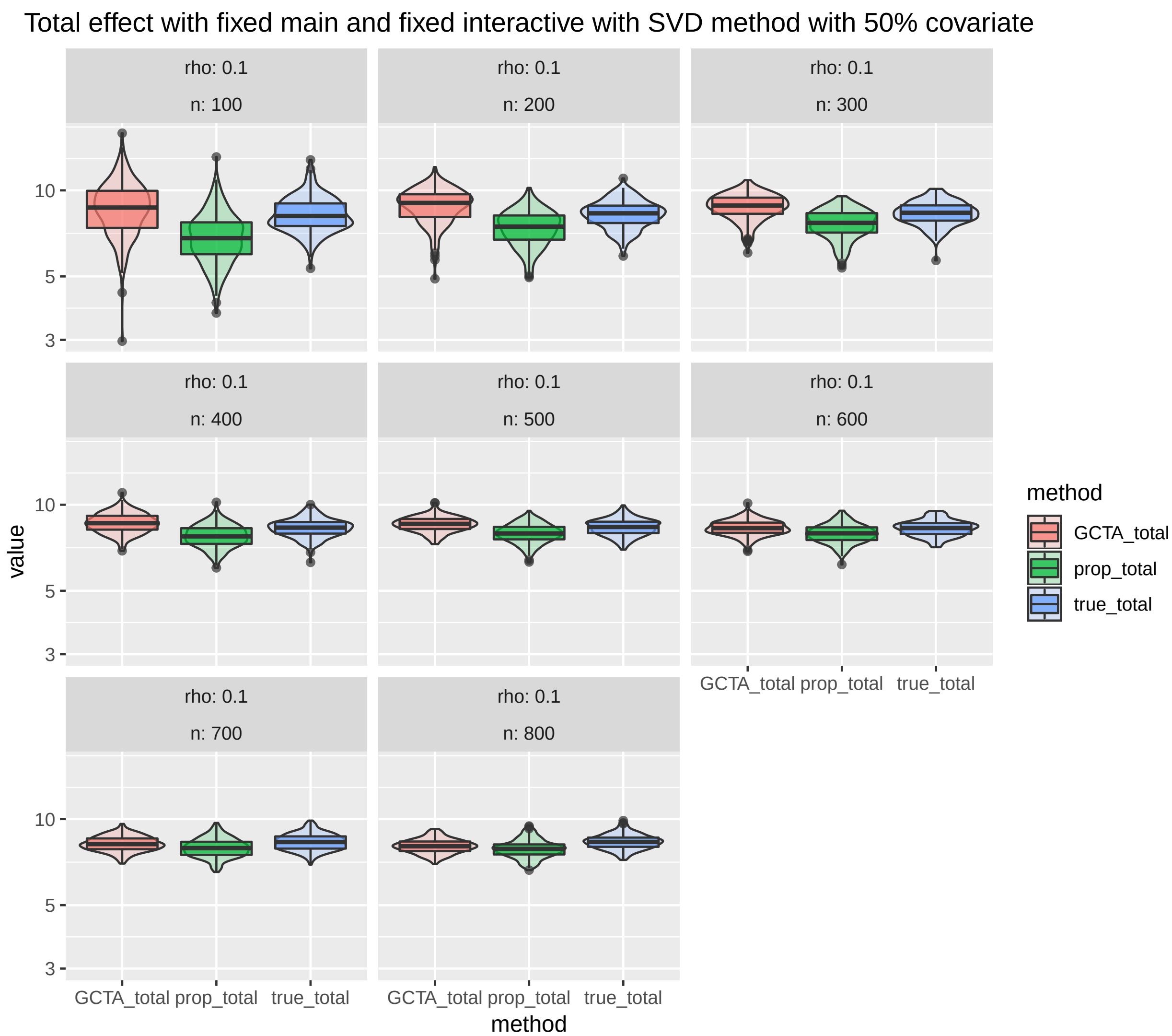


Figure 1: Compound Stymmetry

3.1.3 Autoregression AR(1)

$$T = [T_1 \ldots, T_p], \quad T_i \sim \chi^2_{(1)}, \quad cov(T_i, T_j) = 2\rho^{2|i-j|}, \quad \forall i \neq j, \rho = \{0.1, \ldots, 0.9\}$$

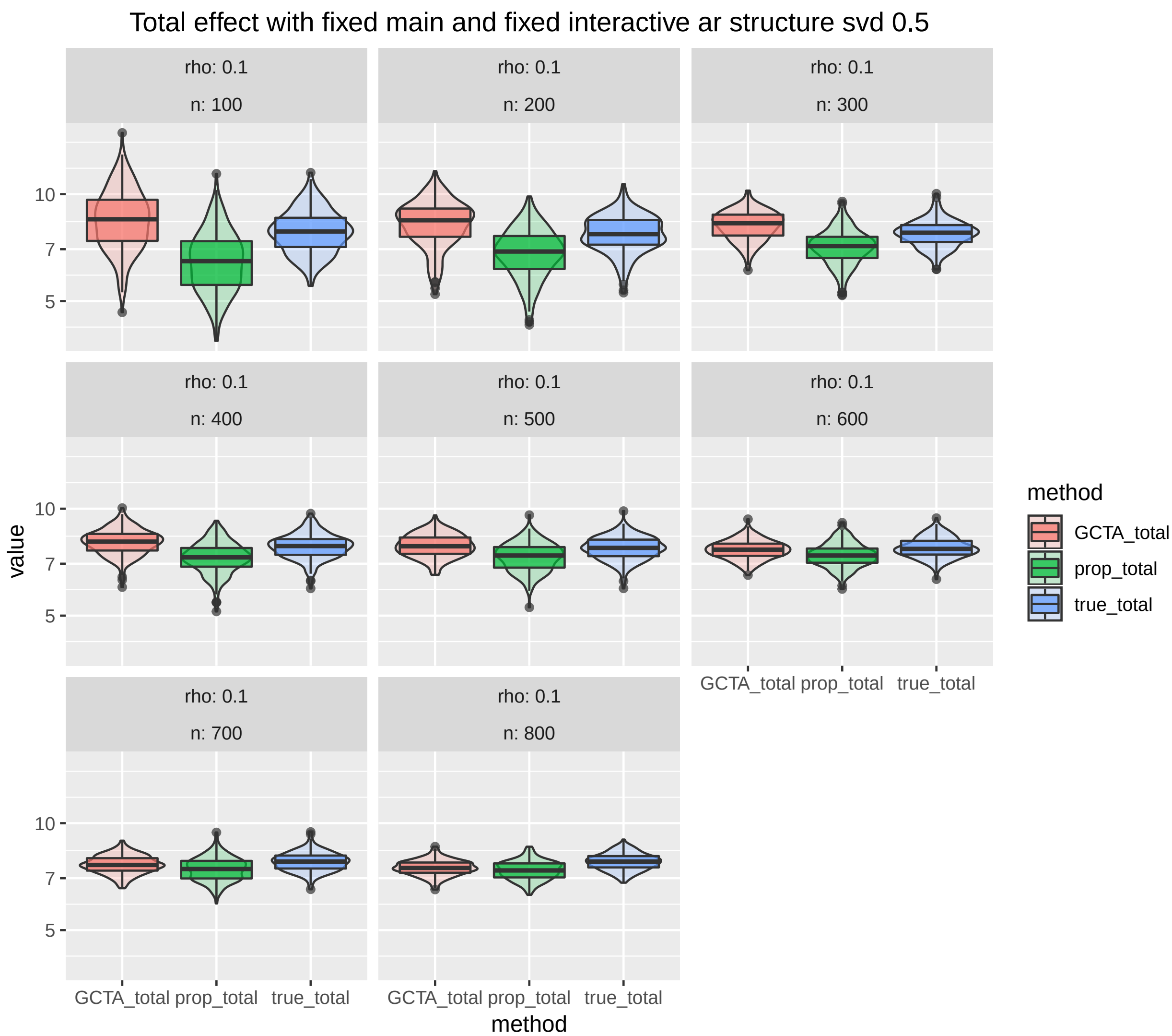


Figure 2: AR(1)

3.1.4 Unstructure

$$T = [T_1 \ldots, T_p], \quad T_i \sim \chi^2_{(1)}, \quad cov(T_i, T_j) = \sigma_{ij}$$

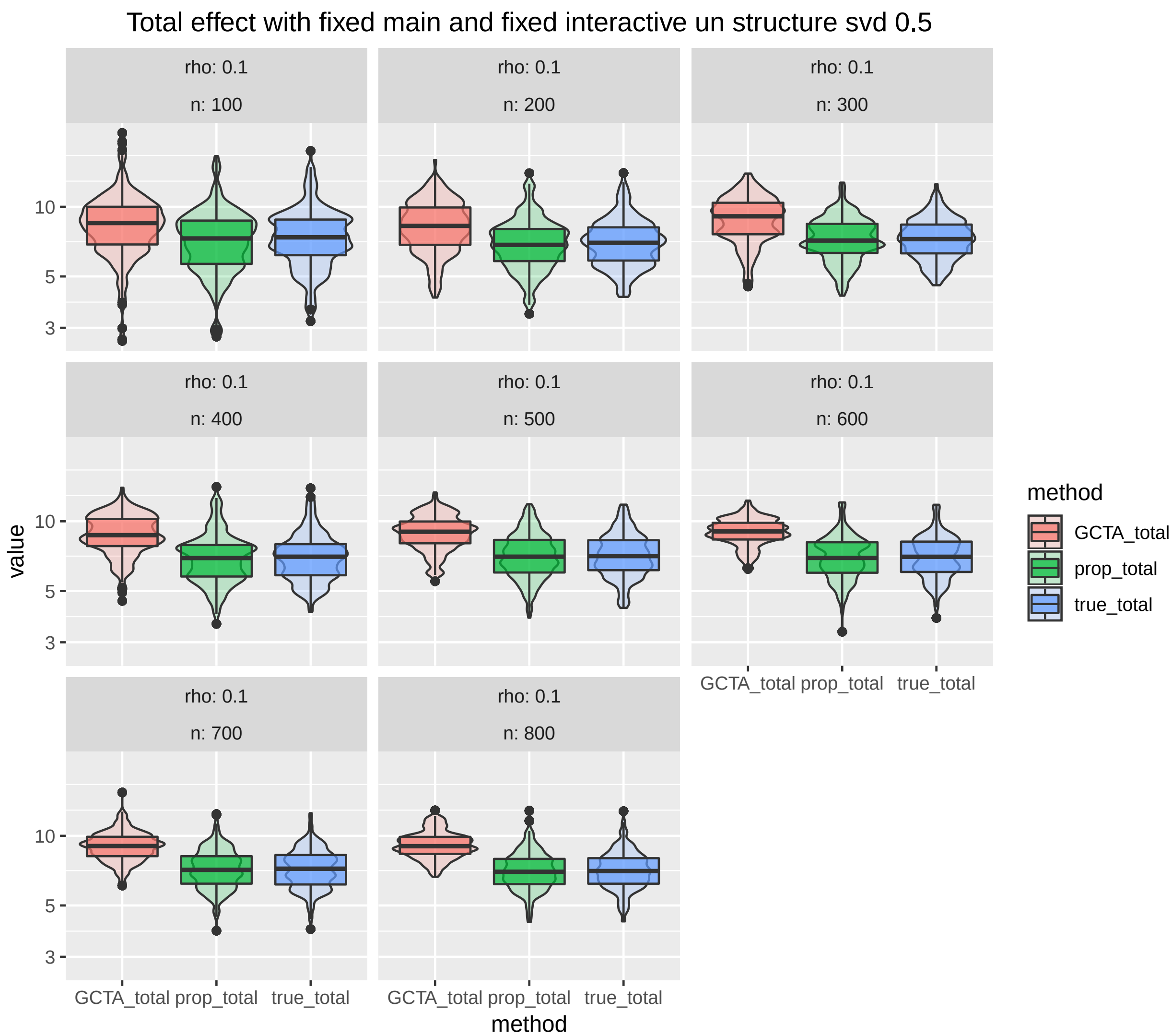


Figure 3: Unstructure

4 PCBs data simulation result

We are using the PCBs data from the

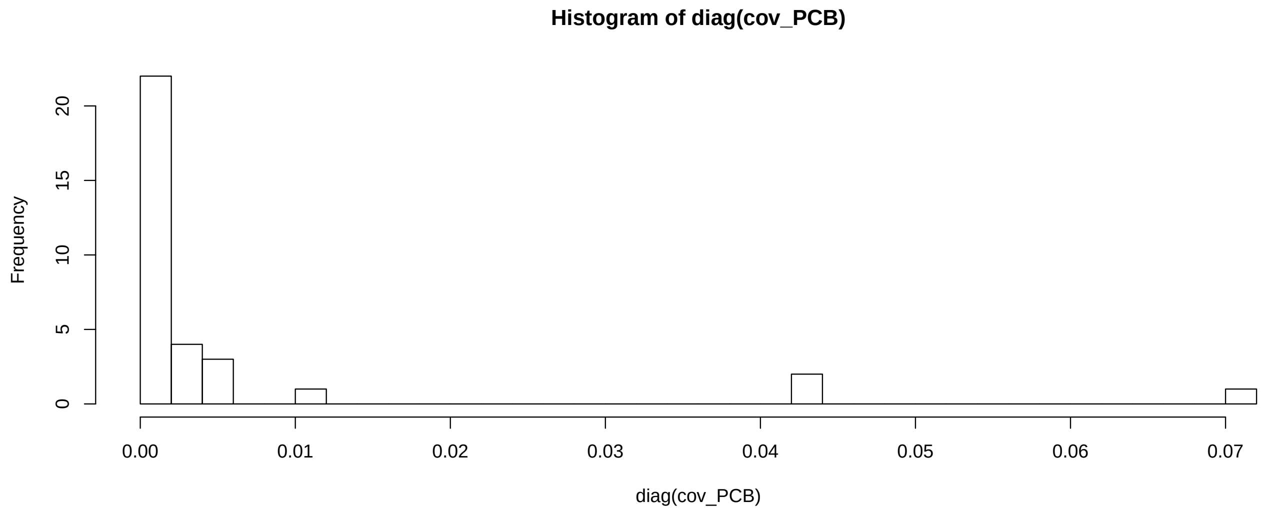
4.1 sample matrix of PCB data

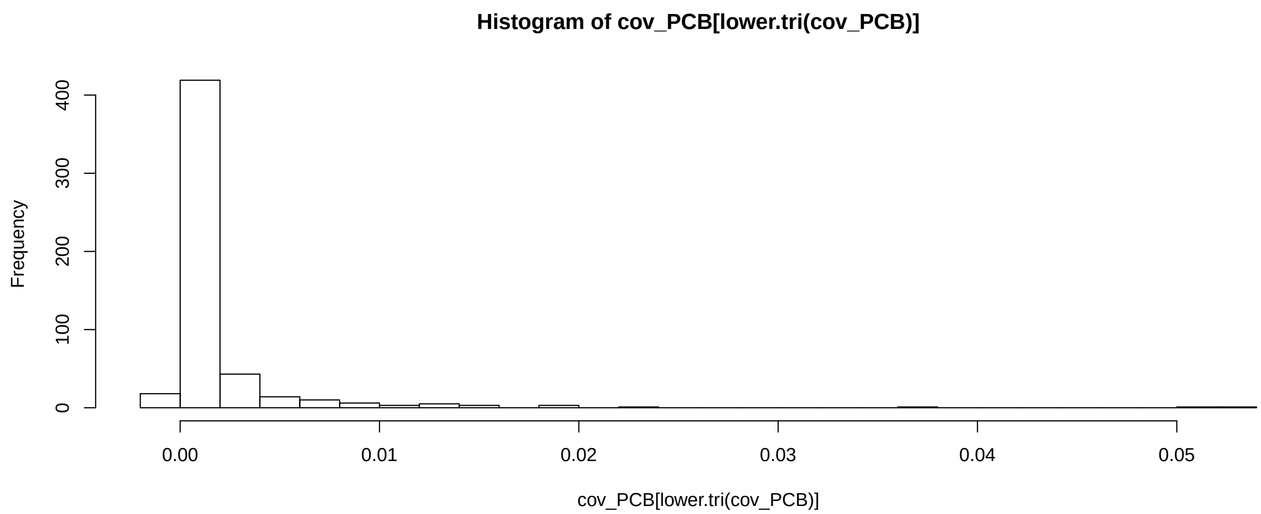
4.1.1 A glimsp of the covariance matrix

Table 1: Covariance of PCB

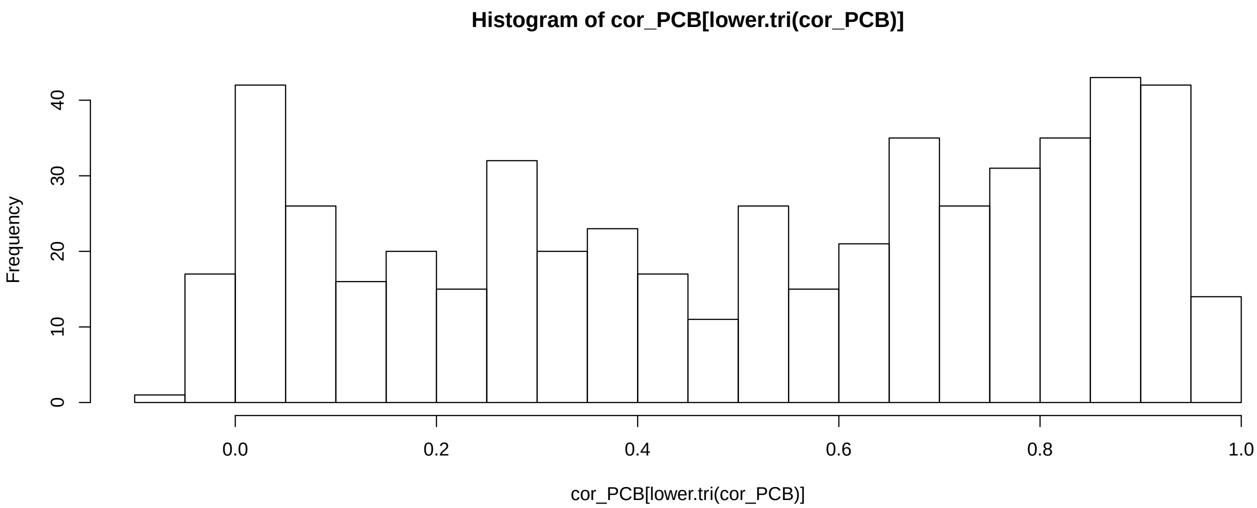
	LBX028	LBX066	LBX074	LBX105	LBX118	LBX156	LBX157	LBX167	LBX044	LBX049
LBX028	1.000	0.678	0.399	0.304	0.311	0.260	0.257	0.280	0.649	0.672
LBX066	0.678	1.000	0.665	0.721	0.694	0.502	0.509	0.629	0.327	0.333
LBX074	0.399	0.665	1.000	0.799	0.856	0.810	0.817	0.880	0.054	0.046
LBX105	0.304	0.721	0.799	1.000	0.974	0.689	0.707	0.840	0.046	0.040
LBX118	0.311	0.694	0.856	0.974	1.000	0.763	0.781	0.906	0.037	0.032
LBX156	0.260	0.502	0.810	0.689	0.763	1.000	0.989	0.890	0.004	0.000
LBX157	0.257	0.509	0.817	0.707	0.781	0.989	1.000	0.908	-0.001	-0.005
LBX167	0.280	0.629	0.880	0.840	0.906	0.890	0.908	1.000	-0.015	-0.019
LBX044	0.649	0.327	0.054	0.046	0.037	0.004	-0.001	-0.015	1.000	0.983
LBX049	0.672	0.333	0.046	0.040	0.032	0.000	-0.005	-0.019	0.983	1.000

4.1.2 Histgram of diagonal and off-diagonal elements of the PCBs'sample covariance





4.1.3 Histgram of off-diagonal elements of the PCBs'sample correlation-coefficient



Based on the correlation coefficient values, it seems that there is no an obvious pattern and the correlations are basically uniformly distributed. Thus, the sample covariance of PCB is more likely to have an unstructure structure.

4.2 Simulation result

One thing about the PCB simulation is that we are using sub-sampling to evaluate the performance of the PCB data.

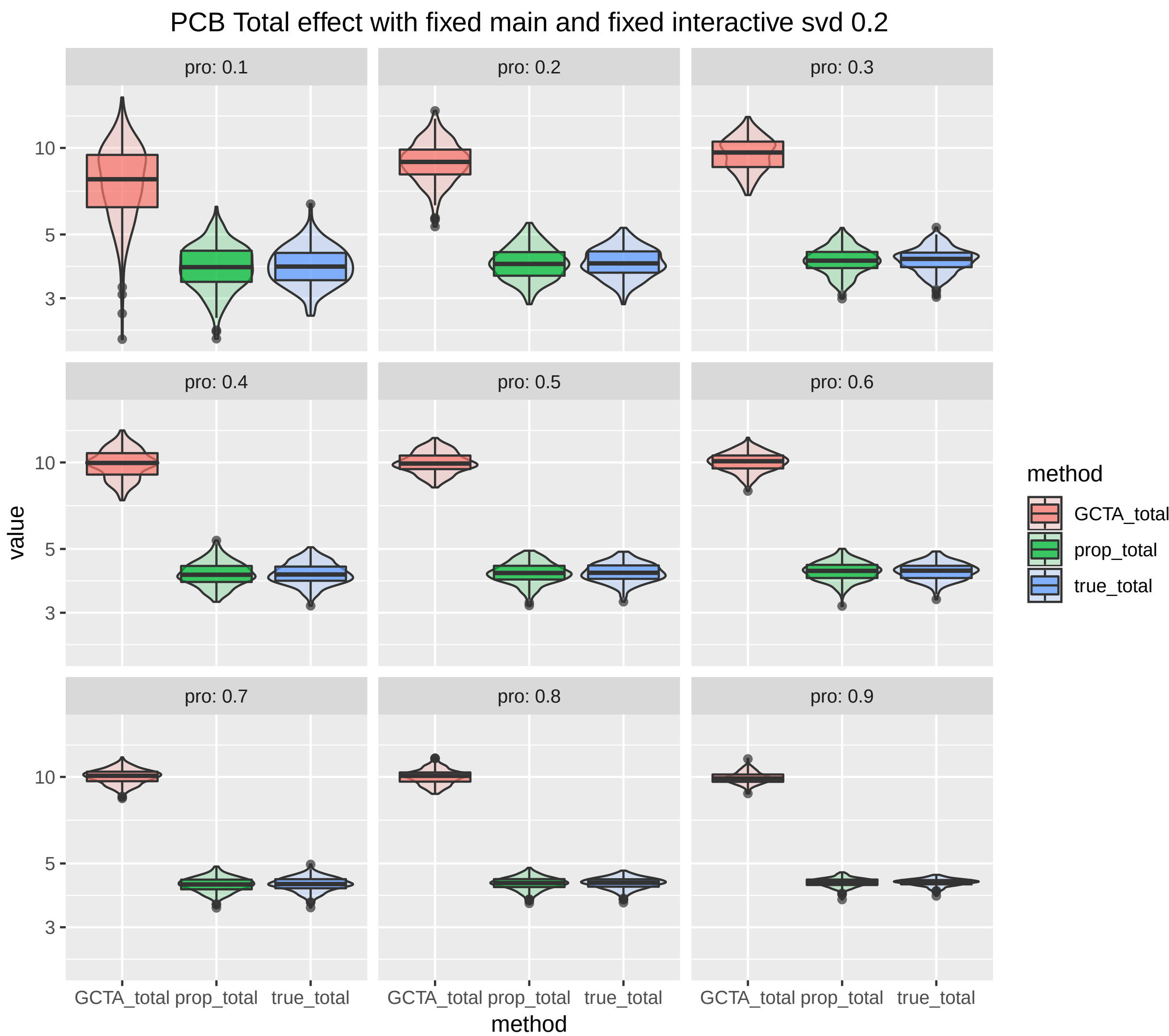


Figure 4: PCB with 0.2

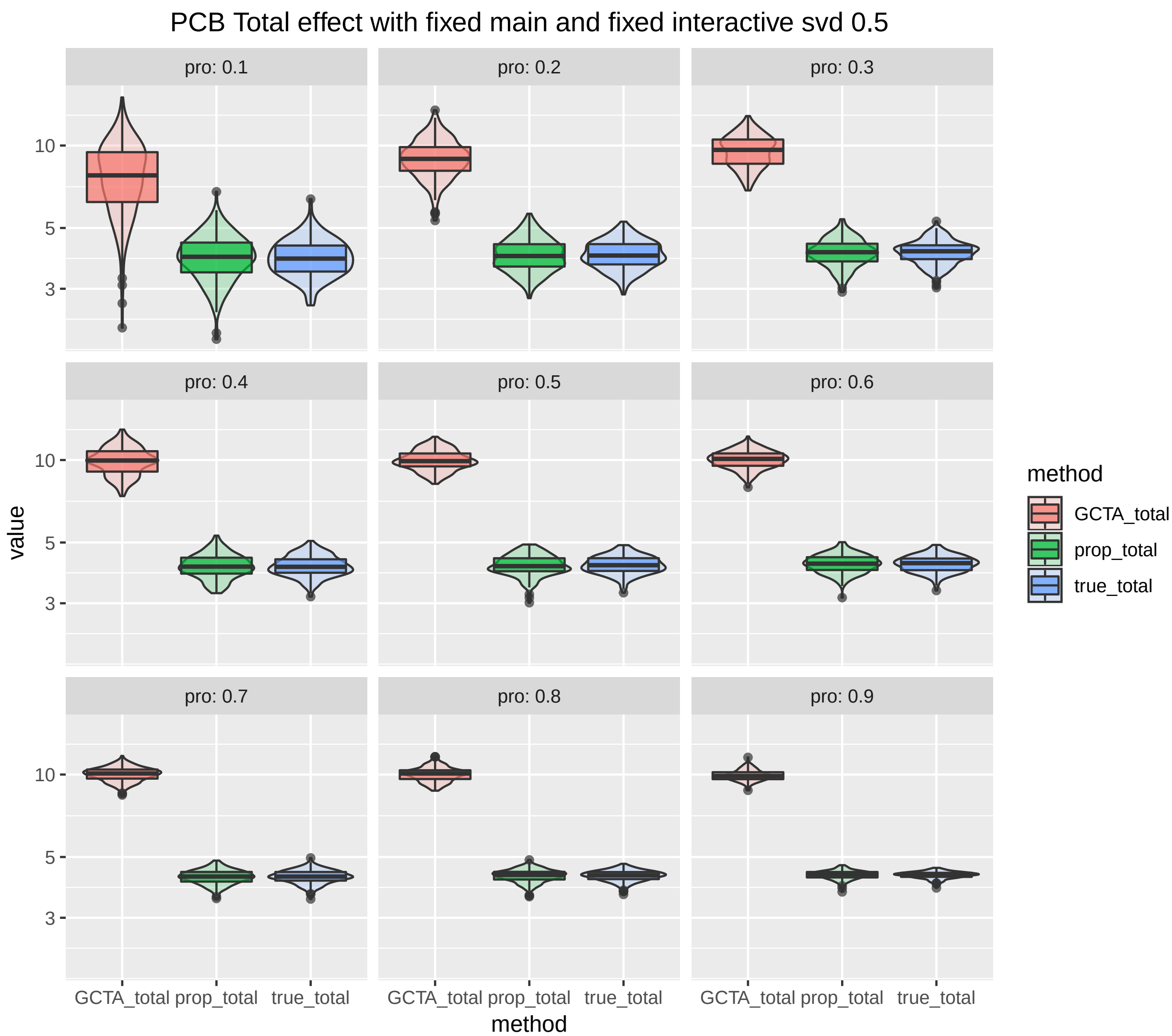


Figure 5: PCB with 0.5

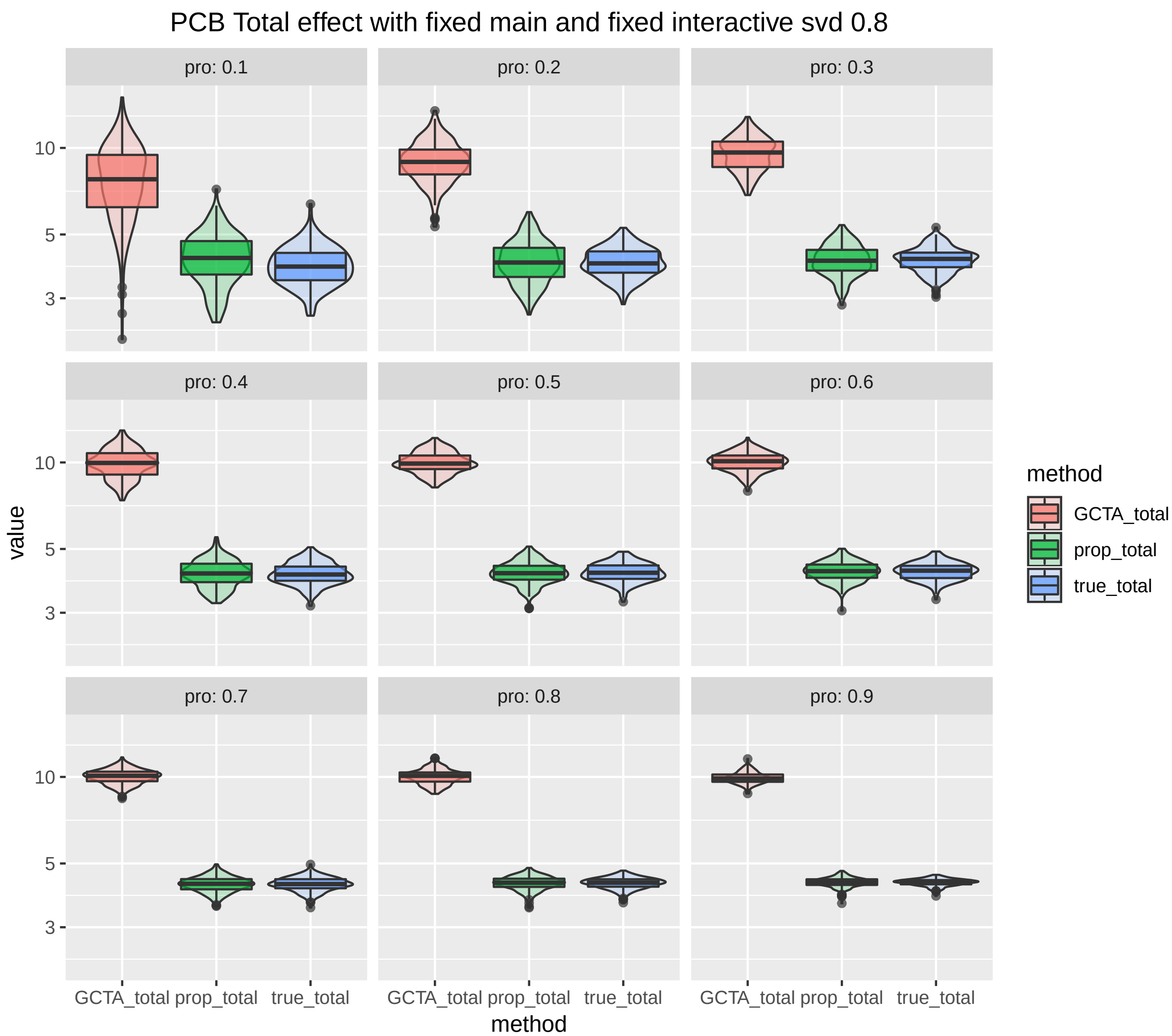


Figure 6: PCB with 0.8