Fused Sparse Structural Equation Models to Jointly Infer Gene Regulatory Network (fssemR)

Xin Zhou

2019-05-31

In this vignette, we introduce the functionality of the `fssemR` package to estimate the differential gene regulatory network by gene expression and genetic perturbation data. To meet the space and time constraints in building this vignette within the `fssemR` package, we are going to simulate gene expression and genetic perturbation data instead of using a real dataset. For this purpose, we will use function `randomFSSEMdata` in `fssemR` to generate simulated data, and then apply fused sparse structural equation model (FSSEM) to estimate the GRNs under two different conditions and their differential GRN. Also, please go to `https://github.com/Ivis4ml/fssemR/tree/master/inst` for more large dataset analysis. In conlcusion, this vignette is composed by three sections as follow,

- Simulating two GRNs and their eQTL effects under two different conditions
- Estimating GRNs from the simulated gene expression data and genetic perturbation data
- Differential GRN Visualization

For user using package `fssemR`, please cite the following article:

Xin Zhou and Xiaodong Cai. Inference of Differential Gene Regulatory Networks Based on Gene Expression and Genetic Perturbation Data. Bioinformatics, submitted.

Simulating two GRNs and their eQTL effects under two different conditions (Acyclic example)

We are going to simulate two GRNs and their corresponding gene expression and genetic perturbation data in the following steps:

1. Load the necessary packages

```
library(fssemR)
library(network)
library(ggnetwork)
library(Matrix)
```

2. Simulate 20 genes expression data from a directed acyclic networks (DAGs) under two conditions, and each gene is simulated having average 3 cis-eQTLs. Also, the genotypes of corresponding eQTLs are generated from F2-cross.

```
n = c(100, 100)     # number of observations in two conditions
p = 20              # number of genes in our simulation
k = 3               # each gene has nonzero 3 cis-eQTL effect
sigma2 = 0.01       # simulated noise variance
prob = 4            # average number of edges connected to each gene
type = "DG"         # `fssemR` also offers simulated ER and directed graph (DG) network
dag  = TRUE         # if DG is simulated, user can select to simulate DAG or DCG
seed = as.numeric(Sys.time())  # any seed acceptable
## seed = 100         # set.seed(100)
set.seed(seed)
data = randomFSSEMdata2(n = n, p = p, k = p * k, sparse = prob / 2, df = 0.3,
                        sigma2 = sigma2, type = type, dag = T)
```

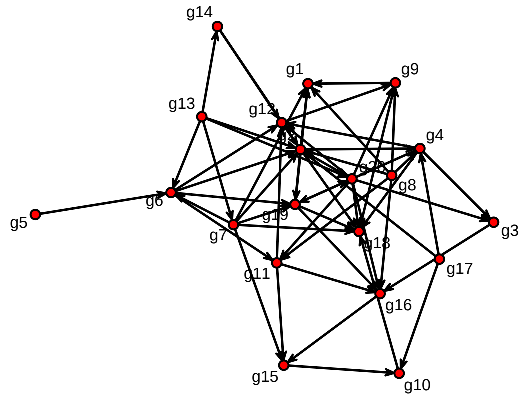

Figure 1: Simulated GRN under condition 1

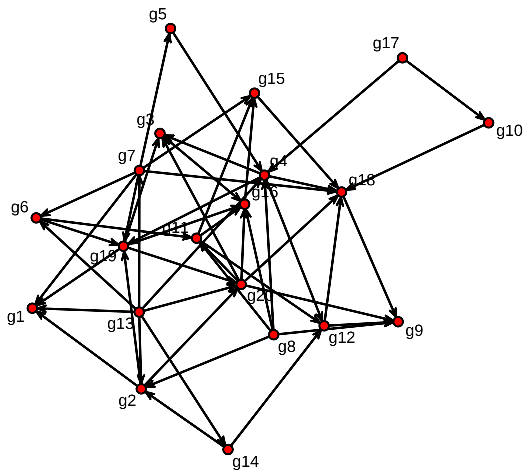

Figure 2: Simulated GRN under condition 2

- Summary of simulated GRNs under two conditions, for simplicity, we named our simulated genes as `g{%d}` and eQTLs as `rs{%d}`.

```
# data$Vars$B[[1]]    ## simulated GRN under condition 1
GRN_1 = network(t(data$Vars$B[[1]]) != 0, matrix.type = "adjacency", directed = TRUE)
> <sparse>[ <logic> ] : .M.sub.i.logical() maybe inefficient
plot(GRN_1, displaylabels = TRUE, label = network.vertex.names(GRN_1), label.cex = 0.5)
```

```
# data$Vars$B[[2]]    ## simulated GRN under condition 2
GRN_2 = network(t(data$Vars$B[[2]]) != 0, matrix.type = "adjacency", directed = TRUE)
> <sparse>[ <logic> ] : .M.sub.i.logical() maybe inefficient
plot(GRN_2, displaylabels = TRUE, label = network.vertex.names(GRN_2), label.cex = 0.5)
```

```
# data$Vars$B[[2]]    ## simulated GRN under condition 2
diffGRN = network(t(data$Vars$B[[2]] - data$Vars$B[[1]]) != 0, matrix.type = "adjacency", directed = TRU
> <sparse>[ <logic> ] : .M.sub.i.logical() maybe inefficient
ecol = 3 - sign(t(data$Vars$B[[2]] - data$Vars$B[[1]]))
plot(diffGRN, displaylabels = TRUE, label = network.vertex.names(GRN_2), label.cex = 0.5, edge.col = ec
```

- Simulated eQTLs's effect for 20 genes.

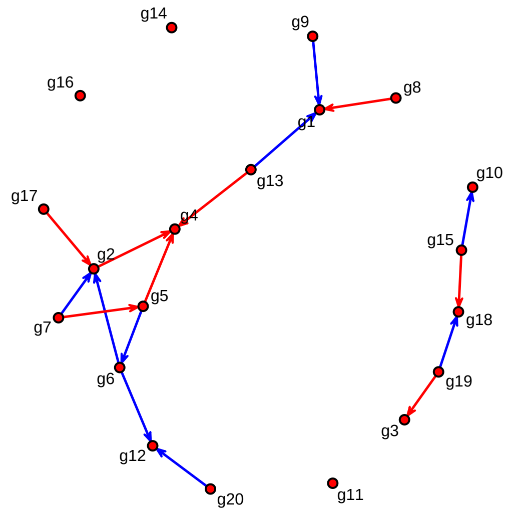

Figure 3: Simulated differential GRN (GRN2 - GRN1), up-regulated are red and down-regulated are blue

```
library(Matrix)
print(Matrix(data$Vars$F, sparse = TRUE))
> 20 x 60 sparse Matrix of class "dgCMatrix"
>     [[ suppressing 31 column names 'rs1', 'rs2', 'rs3' ... ]]
>
> g1  1 . . . . . . . . . . . . . . . . . . . 1 . . . . . . . . . . ......
> g2  . 1 . . . . . . . . . . . . . . . . . . . 1 . . . . . . . . . ......
> g3  . . 1 . . . . . . . . . . . . . . . . . . . 1 . . . . . . . . ......
> g4  . . . 1 . . . . . . . . . . . . . . . . . . . 1 . . . . . . . ......
> g5  . . . . 1 . . . . . . . . . . . . . . . . . . . 1 . . . . . . ......
> g6  . . . . . 1 . . . . . . . . . . . . . . . . . . . 1 . . . . . ......
> g7  . . . . . . 1 . . . . . . . . . . . . . . . . . . . 1 . . . . ......
> g8  . . . . . . . 1 . . . . . . . . . . . . . . . . . . . 1 . . . ......
> g9  . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . 1 . . ......
> g10 . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . 1 . ......
> g11 . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . 1 ......
> g12 . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . ......
> g13 . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . ......
> g14 . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . ......
> g15 . . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . ......
> g16 . . . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . ......
> g17 . . . . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . ......
> g18 . . . . . . . . . . . . . . . . . 1 . . . . . . . . . . . . . ......
> g19 . . . . . . . . . . . . . . . . . . 1 . . . . . . . . . . . . ......
> g20 . . . . . . . . . . . . . . . . . . . 1 . . . . . . . . . . . ......
>
>  .....suppressing columns in show(); maybe adjust 'options(max.print= *, width = *)'
>  ..............................
```

Therefore, the B matrices and F matrix in `data$Vars` are the true values in our simulated model. We then need to estimated the $\hat{B}$ and $\hat{F}$ by the FSSEM algorithm.

Estimating GRNs from the simulated gene expression data and genetic perturbation data

We need to input the gene expression and corresponding genotype data of two conditions into the FSSEM algorithm. They are stored in the `data$Data`.

1. 20 simulated gene expression under two conditions

```
head(data$Data$Y[[1]])
>           [,1]      [,2]       [,3]       [,4]       [,5]       [,6]
> g1  0.9760977  3.280157  2.1872921  0.3619851 -0.8672816  1.3400889
> g2  1.2208697 -3.487021 -1.3699199 -1.7721472 -2.2513582  0.8448976
> g3  5.0250194  4.467497  5.9776772  5.4594733  5.7101969  7.4637000
> g4  4.8747809  9.050986  5.8118360  8.5754132  7.9388685  5.8359289
> g5 -0.6241726  1.222866 -0.7397743  1.3462738  1.4310637  3.3950553
> g6  9.8386216  8.018775  8.5318563  9.9214291  7.8603219 11.4726626
>            [,7]       [,8]      [,9]      [,10]     [,11]      [,12]
> g1 -0.06007814  1.4401248 2.824836  2.6794122  2.104817  0.7095001
> g2 -1.62639039 -0.3805592 1.434141 -2.3581145 -2.324718 -0.3793194
> g3  7.99120311  5.9491710 7.368296  5.3948589  5.994982  5.3970779
> g4  8.20507817  7.9956921 4.685246  8.1858453  9.288694  8.1060762
> g5  1.43602618  1.2106669 2.343406  0.4788932  2.398177  2.4574475
> g6 10.43766489  9.7118186 8.136553  6.9389841  6.293321  8.2636002
>          [,13]      [,14]       [,15]       [,16]     [,17]     [,18]
> g1  1.7030394  0.5420075 -0.06150693 -0.43299104 3.0142115  3.493543
> g2  0.2018353 -1.1248159  2.21118684  0.09892588 0.7261091 -3.735623
> g3  7.1271501  6.7512934  7.59141955  6.03295975 6.3747093  4.704976
> g4  7.4225710  9.1612822  6.06464903 10.00796755 7.7020167  8.822426
> g5  2.4140953  1.4034249  4.35850168  0.51123868 0.3968407  2.272503
> g6 10.7826069  9.0353674 11.24981430  9.46246326 8.6305747  9.012133
>        [,19]      [,20]     [,21]      [,22]     [,23]      [,24]
> g1 2.419437  0.8553368  3.212451  2.2899843 2.1337376  0.9815530
> g2 1.108102 -1.8655815 -1.071996 -0.9241211 0.3130348 -0.6293405
> g3 8.915062  7.5677574  6.567794  4.4553728 7.9024879  6.8696140
> g4 5.152152  7.2024534  9.603355  8.7025243 6.8099365  9.0303878
> g5 3.402322  0.4073160  1.243132  2.3125746 2.2289240  0.3294267
> g6 9.204684  9.6620215  7.769599  9.9751099 6.9084480  6.6332459
>          [,25]      [,26]     [,27]     [,28]     [,29]     [,30]
> g1  0.9556965  1.3344937  1.800771  1.955599  3.607657 0.7256229
> g2 -1.4485288  1.8642239 -3.692008 -1.504077 -3.606287 1.4995444
> g3  3.1514917  7.1995667  5.589482  7.391671  5.872458 5.9643415
> g4 11.4084546  8.6203264 10.206503  6.216795  8.352328 6.9134402
> g5  2.4531732 -0.6280363  1.434647  4.376955 -1.664353 0.1997681
> g6 11.0271746  7.8103842  7.883760 11.168685  9.195421 9.2137787
>          [,31]     [,32]      [,33]      [,34]      [,35]     [,36]
> g1  1.4944364  3.728519  0.8341368  0.4621793  1.5529368  3.220020
> g2 -0.9764435 -4.972410 -0.6673045 -0.9801777 -0.7096311 -1.229632
> g3  7.4604595  6.079548  6.1827948  5.7887738  7.5032070  8.391130
> g4  7.5432347  8.502652  9.7146767  6.8396256  6.8840837  6.315951
> g5  0.2561958  1.404467  2.3920614  3.4015021  1.3460992  3.354361
> g6  8.5118170  9.796167 10.0286830  9.8283466  9.5702675  9.540277
>          [,37]     [,38]      [,39]     [,40]     [,41]      [,42]
> g1  1.0133456  2.394241  1.4398319  2.716171  4.684464  1.7959547
> g2 -0.4578723 -1.180323 -0.4323644 -3.370927 -3.134213 -0.9205142
> g3  8.6538144  6.013106  6.2445852  5.845598  6.092089  6.6760039
```

```
> g4  5.1534815  5.267060  8.0155850  9.753862  8.940387  5.5629937
> g5  1.3977293  2.463838 -0.4674258  1.429501  2.518457  0.3678099
> g6  8.1829279  8.130046  7.1088657  9.552860  7.355086  7.8290540
>         [,43]      [,44]      [,45]      [,46]      [,47]     [,48]
> g1  2.420878 -0.2309699  1.224617 -0.3833169  2.4647371 2.3788808
> g2 -1.419646 -1.6194534 -1.972133  1.3440659 -0.7764676 0.9928242
> g3  5.579931  5.7652067  7.223842  8.0650904  5.9825592 9.3953137
> g4  9.276771  9.7744874  9.704438  9.4345551  8.8685177 6.7102412
> g5  3.314340  0.5109847  3.391222  1.2671696  1.2891608 2.4006752
> g6  8.428489  7.9583969  8.208572 10.0811702  9.9378151 8.2757810
>          [,49]      [,50]     [,51]     [,52]      [,53]      [,54]
> g1 -0.5068195 -0.2117226  2.565463  2.602413  2.3951540  3.3918457
> g2 -3.1469407  3.0206517 -2.006197 -1.858188 -0.5601546 -1.4732951
> g3  7.2681507  8.0259654  5.624332  9.261984  6.6394142  7.5340167
> g4  8.5222898  7.9450000  9.930074  6.268228 10.0168175  9.9619763
> g5  0.4573597  3.3568027  2.433918 -1.812626 -0.6052139 -0.5812035
> g6  8.5365782 10.2665992  9.289761  9.140868  7.9228931 10.0397337
>          [,55]     [,56]      [,57]      [,58]       [,59]      [,60]
> g1  3.0155370  2.793304  2.7031346  1.5202489  2.950077696  2.8864455
> g2 -3.3021131 -3.820901 -0.9546728 -2.6185439  0.005708273 -0.1176116
> g3  6.3763526  4.494906  7.3356112  5.7172929  4.832143741  8.1168144
> g4  8.0738259  9.679904  8.1020862  9.9856052  9.419003677  5.9266029
> g5  0.3368631  1.377956  1.5356442  0.3548704  1.318021403  0.3433108
> g6  7.8134689  7.728710  9.8437045  6.2134227 10.667207612  9.0961530
>          [,61]      [,62]      [,63]     [,64]     [,65]     [,66]
> g1  1.2648885  0.2144400  1.7839042  2.252796  2.879903  2.568347
> g2 -0.5536035  0.5634974 -1.3474481 -1.129597 -3.982432 -2.052396
> g3  8.6043625  7.4673494  8.8458195  6.196949  6.168963  5.824106
> g4  7.6971668  7.9531871  7.2840858  8.653339  9.282867  8.744391
> g5  1.3853949  2.3499334  0.4682828  2.414690  2.237209  2.441593
> g6  8.3254025 10.3035651  7.4606645  8.737341  6.905878  9.110214
>         [,67]     [,68]      [,69]     [,70]     [,71]     [,72]
> g1  2.598734  3.825434 -0.9187388  1.042418 2.0947397  2.212702
> g2 -2.333049 -1.034529  0.2099912 -1.767185 0.2362197 -1.468561
> g3  4.171066  5.739420  5.0617681  5.951058 7.2840722  4.560759
> g4 10.112820  8.128115  7.7248523 10.358478 7.9650044 10.223976
> g5  2.325351  3.518320 -0.5605642  1.477745 1.4234312  1.318723
> g6  8.470288 10.362637  9.6025302  8.142500 9.2073732  7.608832
>            [,73]     [,74]      [,75]      [,76]     [,77]         [,78]
> g1  0.005543756  1.491711 -1.6713920  0.7829452  3.033941 -0.0001732897
> g2 -0.449960284 -1.955804  0.2345872 -1.2021403 -5.020015 -1.6172316397
> g3  6.330458652  7.624640  3.9425449  5.2665056  2.635662  4.1374482163
> g4  9.854276871  7.591505  9.5280170  7.0730231 10.612105 10.4976455071
> g5  3.313702342  3.502113  2.3654667  0.5914084  2.553563  0.3075414167
> g6  8.787483066  6.276079 10.2839735  9.6553531  7.728404  8.5560598533
>         [,79]      [,80]     [,81]      [,82]       [,83]      [,84]
> g1 2.3167151  3.4225571  4.264761  2.3024716  2.07429673  0.8843904
> g2 0.7549955  0.2624841 -2.943200 -1.8750214 -0.08347031 -1.7540726
> g3 7.3534963  5.9063645  4.613259  7.4823350  6.01407728  7.6609298
> g4 8.2935035 10.7617227  9.195153  9.3789274 10.35319709  6.5035201
> g5 1.2718542  1.4121396  1.373006  0.4536185  1.22917009  1.6024044
> g6 9.2242080  6.6102415  9.537454  9.8424414  9.11466654  8.7780403
>         [,85]    [,86]     [,87]      [,88]      [,89]      [,90]
```

```
> g1  1.711411 2.628605 -0.172999  0.2469439  2.8089818  2.2677759
> g2 -0.935841 1.044348  0.573564 -0.2942565 -0.8268095 -1.4231202
> g3  7.090477 7.769330  6.194061  4.6082563  4.0831588  5.0179837
> g4  4.939899 9.058206  8.165928  7.8101536  9.8796227  8.9948700
> g5  1.555555 1.439160  1.410276  2.4002787  0.4181975 -0.5078414
> g6  8.010011 8.492258 11.267156  9.1618475  8.5673730  9.6747740
>         [,91]      [,92]    [,93]        [,94]      [,95]      [,96]
> g1  0.3996202  0.6594901 1.387830 -0.005256082  1.7834190  1.0746590
> g2  0.1854545 -0.3796903 1.624161 -1.022540245  1.3121941 -0.3267415
> g3  6.6691002  6.2368818 6.425699  5.816048271  5.0054151  5.5711552
> g4  8.8117468  7.4380654 8.526928  9.052826658  7.6261928  7.6586312
> g5  1.2648913 -0.6066670 1.158247  2.463439074 -0.7151653  0.4049196
> g6 10.2197992  8.5476012 8.391524 11.077415911  9.5352557  8.9069743
>          [,97]       [,98]     [,99]     [,100]
> g1 -0.06311927  3.28198836  3.201363  2.5240863
> g2 -1.77888611 -0.05285241 -3.574307 -0.7500837
> g3  6.17626285  6.74680043  8.452498  7.6434323
> g4 10.39891051  6.72217768  7.465794  4.9318686
> g5  2.20346589  0.53560938  3.525539  1.3742062
> g6  8.61554539  9.49668740  9.649027  8.1944862
head(data$Data$Y[[2]])
>         [,1]      [,2]       [,3]      [,4]       [,5]      [,6]
> g1 12.421348 13.939514 10.4682981  9.697508 12.4193957  9.582459
> g2 13.559190 14.575359 14.8373537 15.050473 13.4159898 15.889819
> g3  6.090521  5.301937  4.0420869  5.794779  6.5349598  6.781457
> g4 10.356855  7.786950 10.2866636  8.310257  8.0271543  6.469170
> g5  1.805230  1.901621  1.1197743  2.238596  2.1218521  1.253050
> g6 -1.841122 -1.559663 -0.2681079 -1.999376 -0.9709407 -1.605686
>          [,7]       [,8]       [,9]      [,10]     [,11]      [,12]
> g1 13.6811006  9.1353483 12.6092651 10.8875267  8.165272 10.8764397
> g2 13.8835201 13.8565241 14.9350909 16.2381459 14.614421 16.2361596
> g3  5.3884155  3.3603966  6.2768171  4.9354272  5.065812  8.7452382
> g4  9.6467806 10.9240268  9.0857811 12.8019327  8.545310  5.0058203
> g5  0.5206407  0.8613668  4.2269006  2.1770140  2.296569  1.1675054
> g6 -2.1215433 -0.4678253  0.6534172 -0.6883439 -2.195422 -0.6746669
>         [,13]     [,14]      [,15]     [,16]      [,17]     [,18]
> g1 11.2652260 12.064873 11.4571341 10.885178 12.1225892 10.479314
> g2 13.6189473 16.001288 14.3446827 15.150806 14.6849734 14.749356
> g3  7.4440885  3.628977  6.2496081  9.186236  7.9251653  5.665321
> g4  7.6830145 10.056652  9.1299633  7.866091  7.2596891  8.423658
> g5  1.8253531  2.077453  0.7689154  2.991478  0.6955953  3.229889
> g6  0.9551004 -2.214054 -0.2389252  2.362405  1.9915998 -3.528319
>        [,19]     [,20]      [,21]     [,22]      [,23]      [,24]
> g1  9.022478  8.811918  8.7950166 11.379602  9.3461488  7.9673175
> g2 13.819343 15.282754 14.9005132 15.317924 17.8057644 17.2948066
> g3  4.093027  6.545852  6.0859422  5.970871  7.4482095  3.8203089
> g4 11.021589  6.966503 11.2729197 10.326152  9.3216472 12.7325492
> g5  1.918996  2.733993  0.7918018  1.686952  1.4214165  4.9186143
> g6 -3.714206 -1.389584 -3.0758659 -1.199730 -0.2368917  0.6801932
>         [,25]      [,26]     [,27]      [,28]     [,29]     [,30]
> g1  9.2918798 12.4455659 11.012991 11.4462010  9.250983 12.548932
> g2 17.4402205 15.1730066 15.669904 14.3162522 13.557195 15.585853
> g3  4.4315018  7.9641392  6.726400  5.1498886  5.865419  8.031191
```

```
> g4  8.1072411  6.5100669 11.119068  8.4690839  9.450572  7.538238
> g5  0.1302558  2.1375080  3.005959  1.7921840  3.031612  1.551134
> g6 -0.2511320 -0.2358785 -1.260069  0.8929197 -1.422987 -2.844036
>          [,31]      [,32]      [,33]     [,34]      [,35]        [,36]
> g1  8.55799315 10.1005132 11.3598960  9.445973  9.1320388 10.994626334
> g2 15.07048709 16.9208811 12.9725653 14.624693 17.1992348 16.217121404
> g3  3.54552994  8.8505447  6.8702175  5.829938  7.5010352  5.009296662
> g4 10.84798052  8.0787119  7.7357783 10.227254  9.7841394  8.333458410
> g5 -0.07834701  0.8517143  0.2592897  5.090105  1.4002551  1.932854746
> g6 -2.07093267 -1.6243292 -2.4731076 -3.006055 -0.4803651  0.009014963
>         [,37]      [,38]      [,39]      [,40]     [,41]       [,42]
> g1 11.3430692 11.0557017  9.3233797 13.2251969  9.622528 13.17386641
> g2 16.5206885 14.9703991 14.1916000 14.6228009 13.886632 13.84794915
> g3  8.4308565  5.8583730  5.3478740  5.0530378  6.722234  5.37629655
> g4  7.4558399 10.6942897  9.4270747  9.0109927 11.075237  7.18704209
> g5  2.1710422  4.1357176 -0.4929403  4.1585127  2.516151  1.99571787
> g6 -0.7435883  0.3715317 -3.0098080  0.3677519 -1.774071  0.08905287
>         [,43]      [,44]     [,45]       [,46]      [,47]     [,48]
> g1  7.9006344 11.4556259 10.733040 10.78897519 11.0760226  9.176247
> g2 16.0876709 15.4049584 12.905836 15.46223642 14.2956635 15.364301
> g3  7.3914148  4.3257470  4.780283  7.36168647  5.5146810  4.659808
> g4  8.1639488  8.8990320  8.965596  7.61468683  7.8063206  6.751693
> g5  3.6770080  2.0742942  2.306125  0.63014362  1.7282007  2.360323
> g6  0.2077332 -0.8567174 -3.040837 -0.05533534  0.2165488 -1.915598
>         [,49]      [,50]     [,51]      [,52]      [,53]      [,54]
> g1 11.4848374  9.2971305 11.704573 12.8073798 11.1575882 10.9992968
> g2 16.5691776 15.5193413 15.853545 14.1628827 15.4909653 16.1466768
> g3  4.7595659  6.5234017  5.604829  7.2659018  5.8683784  8.5463680
> g4  9.0126863  9.5807771  8.164567  6.4320727  8.0273308  6.2529782
> g5  0.2834217 -0.3437727  2.194745  0.1938738 -0.2039987  0.5548419
> g6 -1.4099215 -0.7180651 -2.345866 -1.3105363 -0.1204937 -2.2245128
>        [,55]     [,56]      [,57]      [,58]     [,59]     [,60]     [,61]
> g1 10.099725  9.301491 11.0158639 11.0605795 10.625313  8.698852 10.101893
> g2 15.752778 15.237308 14.8237823 17.1756853 14.532018 11.695570 14.696353
> g3  6.560473  6.576924  6.3127501  7.1680744  6.414323  3.820331  4.034472
> g4  8.343541 10.978782  8.6941060  7.9990022  7.051228  8.276259  8.490699
> g5  3.769293  1.848182  2.5710955  0.9437620  3.126032  1.456099  2.127115
> g6 -1.039436 -1.520205 -0.4260627 -0.9189164 -2.711519 -1.542206 -3.393978
>        [,62]      [,63]      [,64]     [,65]      [,66]      [,67]
> g1 11.339780 10.6755771 10.5291608  9.625290 10.3561244 13.8731646
> g2 13.962761 14.6892633 13.6352477 17.203017 15.4539390 14.1091206
> g3  5.982470  8.2507536  6.6127652  8.452573  8.0751739  6.3592704
> g4  7.845071  9.4805531  9.8201559  8.412066  7.9027743  7.9929315
> g5  3.454509  3.1911334  2.2574537  1.071078  0.1145811  1.1933084
> g6  1.132689 -0.1414293  0.2959745 -1.912952  0.2367002 -0.9669737
>        [,68]     [,69]     [,70]     [,71]      [,72]       [,73]
> g1  9.882352  7.785578  7.959323 10.973299  8.9347406 12.91760392
> g2 15.278129 14.024404 18.272291 16.704438 12.7762678 16.07001629
> g3  7.974342  2.486505  5.417267  5.833347  5.0192988 10.25148789
> g4  6.883199  9.959929 10.148798  8.085192 10.2001584  6.28870351
> g5  4.164372  1.843450  2.372003  2.213604  2.6732252 -0.02300952
> g6  1.005863 -2.624677 -1.609583 -1.330861  0.4890438 -0.80285660
>         [,74]      [,75]     [,76]      [,77]      [,78]     [,79]
```

```
> g1  9.7357154 12.5256728 10.784357  9.4311875 14.1469583  7.572061
> g2 16.4499091 14.0331695 15.239671 12.5780808 14.7215939 16.822506
> g3  7.9768313  3.5955725  7.112142  5.7475618  6.6027561  5.361782
> g4  8.6020843  7.6883514  8.762813  8.3260277  8.4257074  7.972074
> g5  3.7432485  0.4495451  1.680403  0.3041768 -0.1648785  2.287273
> g6  0.1384536 -1.6054919 -1.386435 -3.6444515 -1.7618055 -1.977579
>         [,80]       [,81]     [,82]      [,83]      [,84]      [,85]
> g1  8.0846816 11.71426995  8.944367  9.9875454  9.6176522 11.5980369
> g2 16.3368002 14.02493425 14.950336 16.4402813 16.3070894 17.0841395
> g3  8.3795066  5.91787397  4.799900  6.0705790  5.0958480  8.1067515
> g4  8.5010159  7.46307380  9.212135  7.8553893 10.4574008  6.8106768
> g5  1.3918909  1.96759108  4.671450  2.0928477  2.2146322  3.2726891
> g6 -0.6872934  0.05852658 -1.601683  0.5868051 -0.1118209 -0.2302121
>         [,86]     [,87]      [,88]     [,89]     [,90]      [,91]
> g1 10.2113250 11.298278  8.7338892 10.251772 11.381679 10.0230808
> g2 14.2704540 14.107496 14.4731119 14.581469 15.873335 15.3879010
> g3  7.1482801  3.227418  4.9319964  7.500260  4.232761  6.6279882
> g4  9.4806355  9.016963  9.8188625  9.354155 11.012975  9.5812169
> g5 -0.8165853  3.947325  0.8247575  2.944737  2.765676  0.9791880
> g6 -2.9941091 -1.925497 -1.2249160 -1.207771 -1.501459 -0.5525695
>         [,92]     [,93]     [,94]     [,95]      [,96]     [,97]     [,98]
> g1 12.8886110 10.375624 11.061010 10.877344 11.6393872 12.523269  9.766160
> g2 15.1743525 14.540918 17.163542 16.754136 12.7183740 17.509022 18.102682
> g3  6.8960876  4.134341  3.879834  6.083747  4.4833611  5.592559  7.047250
> g4  7.5040503  9.797249  8.912298  7.739183  9.6259776  8.488651  6.973501
> g5  1.7151115  1.290577  2.754755  4.360778  0.1345944  2.128044  1.640061
> g6 -0.2013386 -2.868762 -3.283933 -1.200672 -2.0015603 -2.125241 -1.775072
>        [,99]    [,100]
> g1  9.092923 11.484572
> g2 15.250683 14.600792
> g3  5.246449  5.628048
> g4  7.903881  9.029427
> g5  2.691186  2.629498
> g6  1.344443 -2.722771
```

2. 60 corresponding cis-eQTLs' genotype under two conditions

```
head(data$Data$X[[1]] - 1)
>     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
> rs1    1    1    0    1    0    0    1    1    1     1     1     0     1
> rs2    1    2    0    2    0    2    2    1    2     1     1     1     2
> rs3    0    1    2    1    0    1    2    1    0     1     1     1     1
> rs4    0    1    1    1    1    0    1    2    1     1     2     2     1
> rs5    1    1    0    1    2    2    1    2    1     1     2     2     1
> rs6    2    1    1    1    0    2    1    1    1     1     0     1     2
>     [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24]
> rs1     0     1     1     1     1     0     0     1     1     1     1
> rs2     1     1     1     2     1     1     1     2     2     2     2
> rs3     2     1     0     1     0     1     1     2     1     0     1
> rs4     1     1     2     1     1     1     1     2     0     1     1
> rs5     2     2     0     1     2     2     0     0     2     1     1
> rs6     0     0     2     1     1     1     2     1     1     1     0
>     [,25] [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35]
> rs1     1     1     2     0     1     0     1     1     1     0     0
```

```
> rs2     1     2     1     1     1     1     2     0     0     1     2
> rs3     0     1     0     2     2     1     2     1     1     1     2
> rs4     2     2     1     0     0     1     1     0     2     0     0
> rs5     2     0     0     2     0     0     1     0     1     1     1
> rs6     2     1     1     2     2     1     0     2     1     0     1
>      [,36] [,37] [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46]
> rs1     0     0     1     1     2     2     0     1     0     0     1
> rs2     0     1     1     1     1     0     0     1     1     2     2
> rs3     2     1     0     1     1     2     1     1     0     1     1
> rs4     1     0     0     2     1     2     1     2     1     2     1
> rs5     1     1     1     0     2     2     1     2     1     1     1
> rs6     0     0     1     1     2     0     2     1     0     2     1
>      [,47] [,48] [,49] [,50] [,51] [,52] [,53] [,54] [,55] [,56] [,57]
> rs1     0     1     2     1     1     1     0     2     2     1     1
> rs2     1     1     1     2     1     1     2     1     0     0     2
> rs3     2     2     2     2     1     1     0     1     2     0     2
> rs4     0     2     2     2     1     1     1     1     1     1     1
> rs5     1     1     0     2     1     0     0     1     1     2     1
> rs6     2     0     0     1     1     1     1     1     1     1     1
>      [,58] [,59] [,60] [,61] [,62] [,63] [,64] [,65] [,66] [,67] [,68]
> rs1     1     2     2     1     0     1     0     1     2     2     2
> rs2     0     1     1     1     2     1     1     0     0     1     1
> rs3     1     1     1     1     1     2     2     1     2     1     2
> rs4     2     2     0     0     1     0     2     2     2     1     1
> rs5     1     1     2     1     2     0     1     2     2     2     2
> rs6     0     1     2     0     2     1     2     1     1     0     2
>      [,69] [,70] [,71] [,72] [,73] [,74] [,75] [,76] [,77] [,78] [,79]
> rs1     0     1     2     2     2     1     0     0     2     0     1
> rs2     0     0     1     1     1     1     2     1     1     1     2
> rs3     0     2     2     1     2     2     1     1     1     0     1
> rs4     1     1     0     2     0     0     1     0     1     2     0
> rs5     1     1     1     1     1     2     1     2     1     2     1
> rs6     2     2     2     1     2     0     2     2     1     2     1
>      [,80] [,81] [,82] [,83] [,84] [,85] [,86] [,87] [,88] [,89] [,90]
> rs1     1     1     1     2     0     0     1     1     0     1     1
> rs2     1     1     0     0     0     1     2     1     1     1     1
> rs3     0     1     2     0     1     0     2     2     0     0     0
> rs4     2     2     1     1     0     0     1     1     0     1     1
> rs5     1     0     1     2     2     1     0     2     2     1     0
> rs6     1     1     1     0     1     1     2     2     1     0     1
>      [,91] [,92] [,93] [,94] [,95] [,96] [,97] [,98] [,99] [,100]
> rs1     1     0     2     0     2     2     0     1     1      2
> rs2     1     1     2     0     2     1     2     1     0      1
> rs3     2     1     1     1     0     1     1     1     1      1
> rs4     2     1     1     1     0     1     0     0     1      1
> rs5     1     0     2     1     0     1     2     0     2      2
> rs6     2     2     1     1     1     2     2     2     0      2
head(data$Data$X[[2]] - 1)
>      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
> rs1    1    1    1    0    1    0    1    1    2     2     0     1     2
> rs2    2    1    1    1    0    1    1    0    1     1     1     0     1
> rs3    1    0    1    0    2    0    1    1    1     0     1     1     1
> rs4    2    1    2    2    1    1    2    2    0     2     1     1     1
```

```
> rs5    1    1    0    1    0    1    0    0    1     1     1     0     2
> rs6    1    0    2    0    1    1    0    1    2     0     1     2     2
>      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24]
> rs1      2     2     2     1     1     1     0     0     1     0     0
> rs2      2     0     0     2     0     1     2     0     1     2     2
> rs3      1     2     2     2     1     1     1     2     1     0     0
> rs4      1     2     2     0     0     1     0     2     2     2     2
> rs5      1     2     1     1     1     1     1     0     0     1     1
> rs6      0     1     1     2     0     0     0     1     1     1     1
>      [,25] [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35]
> rs1      1     2     1     2     1     1     1     0     2     1     0
> rs2      1     1     1     0     1     1     0     2     1     0     1
> rs3      1     2     2     1     2     0     0     2     1     1     1
> rs4      1     0     2     1     1     1     2     1     1     2     2
> rs5      0     1     1     1     2     0     1     0     0     2     0
> rs6      1     0     1     1     2     0     1     1     0     0     1
>      [,36] [,37] [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46]
> rs1      2     1     1     1     2     1     2     0     0     1     1
> rs2      1     1     1     1     1     1     1     1     0     1     0
> rs3      1     2     1     0     1     1     0     1     0     1     2
> rs4      2     1     1     0     2     2     1     1     1     1     1
> rs5      0     1     1     1     2     2     2     2     1     0     0
> rs6      1     0     1     1     0     1     2     2     2     0     2
>      [,47] [,48] [,49] [,50] [,51] [,52] [,53] [,54] [,55] [,56] [,57]
> rs1      2     0     0     0     2     1     0     1     1     1     2
> rs2      0     1     0     2     1     1     1     1     2     1     1
> rs3      1     0     1     2     0     1     1     2     1     0     1
> rs4      1     0     2     2     0     1     0     0     1     1     2
> rs5      0     1     0     1     2     1     1     0     1     2     0
> rs6      1     1     1     1     0     1     2     1     0     1     0
>      [,58] [,59] [,60] [,61] [,62] [,63] [,64] [,65] [,66] [,67] [,68]
> rs1      0     0     0     1     1     0     1     0     0     2     1
> rs2      2     1     0     1     0     2     1     1     2     0     0
> rs3      1     2     1     0     1     2     1     2     1     0     1
> rs4      0     0     0     1     1     1     2     1     1     1     1
> rs5      1     2     0     0     2     1     1     1     0     0     2
> rs6      1     0     1     0     1     1     1     0     1     0     1
>      [,69] [,70] [,71] [,72] [,73] [,74] [,75] [,76] [,77] [,78] [,79]
> rs1      0     0     2     0     1     0     1     1     0     2     0
> rs2      0     2     2     1     1     2     1     1     1     0     1
> rs3      0     0     1     1     2     0     0     1     1     1     1
> rs4      1     1     2     2     1     1     1     1     1     2     1
> rs5      1     2     0     2     0     2     0     2     0     0     1
> rs6      2     1     1     1     1     1     0     2     0     1     1
>      [,80] [,81] [,82] [,83] [,84] [,85] [,86] [,87] [,88] [,89] [,90]
> rs1      1     1     0     0     1     2     1     1     1     2     1
> rs2      2     0     0     1     2     1     1     1     0     0     1
> rs3      2     2     0     1     1     1     2     0     2     1     2
> rs4      0     0     1     2     2     1     2     1     1     1     2
> rs5      1     1     2     1     1     2     0     2     1     0     2
> rs6      0     1     0     2     2     1     1     0     1     1     1
>      [,91] [,92] [,93] [,94] [,95] [,96] [,97] [,98] [,99] [,100]
> rs1      1     2     1     2     2     1     2     1     0      2
```

```
> rs2     1     0     1     2     2     0     2     2     1      0
> rs3     2     1     0     0     0     1     0     1     0      2
> rs4     1     2     2     1     1     2     1     0     2      0
> rs5     0     2     1     1     2     1     0     1     1      1
> rs6     1     2     0     1     1     2     0     0     2      1
```

3. `data$Data$Sk` stores each gene's cis-eQTL's indices. In real data application, we recommend to use package `MatrixEQTL` to search the significant cis-eQTLs for genes of interested and build `Sk` for your research

```
head(data$Data$Sk)
> $g1
> [1]  1 21 41
>
> $g2
> [1]  2 22 42
>
> $g3
> [1]  3 23 43
>
> $g4
> [1]  4 24 44
>
> $g5
> [1]  5 25 45
>
> $g6
> [1]  6 26 46
```

Initialization of `fssemR` by ridge regression

We implement our fssemR by the observed gene expression data and genetic perturbations data that stored in `data$Data`, and it is initialized by ridge regression, the l_2 norm penalty's hyperparameter γ is selected by 5-fold cross-validation.

```
Xs  = data$Data$X     ## eQTL's genotype data
Ys  = data$Data$Y     ## gene expression data
Sk  = data$Data$Sk    ## cis-eQTL indices
gamma = cv.multiRegression(Xs, Ys, Sk, ngamma = 50, nfold = 5, n = data$Vars$n,
                           p = data$Vars$p, k = data$Vars$k)
>  [1] 18.664614 18.458488 18.220446 17.947535 17.637175 17.287385 16.896974
>  [8] 16.465754 15.994656 15.485792 14.942395 14.368669 13.769546 13.150417
> [15] 12.516825 11.874254 11.227941 10.582824  9.943541  9.314503  8.699993
> [22]  8.104269  7.531607  6.986328  6.472724  5.994936  5.556745  5.161319
> [29]  4.810901  4.506552  4.247913  4.033143  3.859011  3.721187  3.614656
> [36]  3.534179  3.474701  3.431658  3.401141  3.379947  3.365545  3.355994
> [43]  3.349849  3.346048  3.343831  3.342662  3.342166  3.342090  3.342262
> [50]  3.342570
fit0   = multiRegression(data$Data$X, data$Data$Y, data$Data$Sk, gamma, trans = FALSE,
                         n = data$Vars$n, p = data$Vars$p, k = data$Vars$k)
```

Run fssemR algorithm for data

Then, we chose the `fit0` object from ridge regression as intialization, and implement the `fssemR` algorithm, BIC is used to select optimal hyperparameters λ, ρ, where `nlambda` is the number of candidate lambda values for l_1 regularized term, and `nrho` is the number of candidate rho values for fused lasso regularized term.

```
fitOpt <- opt.multiFSSEMiPALM2(Xs = Xs, Ys = Ys, Bs = fit0$Bs, Fs = fit0$Fs, Sk = Sk,
                               sigma2 = fit0$sigma2, nlambda = 10, nrho = 10,
                               p = data$Vars$p, q = data$Vars$k, wt = TRUE)
> FSSEM@lambda = 114.801072, rho = 0.000000
> FSSEM@lambda = 114.801072, rho = 0.000000
> FSSEM@lambda = 114.801072, rho = 0.000000
> FSSEM@lambda = 114.801072, rho = 0.000000
> FSSEM@lambda = 114.801072, rho = 0.000000
> FSSEM@lambda = 114.801072, rho = 0.000000
> FSSEM@lambda = 114.801072, rho = 0.000000
> FSSEM@lambda = 114.801072, rho = 0.000000
> FSSEM@lambda = 114.801072, rho = 0.000000
> FSSEM@lambda = 114.801072, rho = 0.000000
> FSSEM@lambda = 53.285937, rho = 0.551229
> FSSEM@lambda = 53.285937, rho = 0.255858
> FSSEM@lambda = 53.285937, rho = 0.118759
> FSSEM@lambda = 53.285937, rho = 0.055123
> FSSEM@lambda = 53.285937, rho = 0.025586
> FSSEM@lambda = 53.285937, rho = 0.011876
> FSSEM@lambda = 53.285937, rho = 0.005512
> FSSEM@lambda = 53.285937, rho = 0.002559
> FSSEM@lambda = 53.285937, rho = 0.001188
> FSSEM@lambda = 53.285937, rho = 0.000551
> FSSEM@lambda = 24.733141, rho = 5.749870
> FSSEM@lambda = 24.733141, rho = 2.668853
> FSSEM@lambda = 24.733141, rho = 1.238772
> FSSEM@lambda = 24.733141, rho = 0.574987
> FSSEM@lambda = 24.733141, rho = 0.266885
> FSSEM@lambda = 24.733141, rho = 0.123877
> FSSEM@lambda = 24.733141, rho = 0.057499
> FSSEM@lambda = 24.733141, rho = 0.026689
> FSSEM@lambda = 24.733141, rho = 0.012388
> FSSEM@lambda = 24.733141, rho = 0.005750
> FSSEM@lambda = 11.480107, rho = 5.967663
> FSSEM@lambda = 11.480107, rho = 2.769944
> FSSEM@lambda = 11.480107, rho = 1.285694
> FSSEM@lambda = 11.480107, rho = 0.596766
> FSSEM@lambda = 11.480107, rho = 0.276994
> FSSEM@lambda = 11.480107, rho = 0.128569
> FSSEM@lambda = 11.480107, rho = 0.059677
> FSSEM@lambda = 11.480107, rho = 0.027699
> FSSEM@lambda = 11.480107, rho = 0.012857
> FSSEM@lambda = 11.480107, rho = 0.005968
> FSSEM@lambda = 5.328594, rho = 391.311262
> FSSEM@lambda = 5.328594, rho = 181.630598
> FSSEM@lambda = 5.328594, rho = 84.305456
> FSSEM@lambda = 5.328594, rho = 39.131126
> FSSEM@lambda = 5.328594, rho = 18.163060
```

```
> FSSEM@lambda = 5.328594, rho = 8.430546
> FSSEM@lambda = 5.328594, rho = 3.913113
> FSSEM@lambda = 5.328594, rho = 1.816306
> FSSEM@lambda = 5.328594, rho = 0.843055
> FSSEM@lambda = 5.328594, rho = 0.391311
> FSSEM@lambda = 2.473314, rho = 433.950557
> FSSEM@lambda = 2.473314, rho = 201.422006
> FSSEM@lambda = 2.473314, rho = 93.491813
> FSSEM@lambda = 2.473314, rho = 43.395056
> FSSEM@lambda = 2.473314, rho = 20.142201
> FSSEM@lambda = 2.473314, rho = 9.349181
> FSSEM@lambda = 2.473314, rho = 4.339506
> FSSEM@lambda = 2.473314, rho = 2.014220
> FSSEM@lambda = 2.473314, rho = 0.934918
> FSSEM@lambda = 2.473314, rho = 0.433951
> FSSEM@lambda = 1.148011, rho = 401.946254
> FSSEM@lambda = 1.148011, rho = 186.566924
> FSSEM@lambda = 1.148011, rho = 86.596695
> FSSEM@lambda = 1.148011, rho = 40.194625
> FSSEM@lambda = 1.148011, rho = 18.656692
> FSSEM@lambda = 1.148011, rho = 8.659670
> FSSEM@lambda = 1.148011, rho = 4.019463
> FSSEM@lambda = 1.148011, rho = 1.865669
> FSSEM@lambda = 1.148011, rho = 0.865967
> FSSEM@lambda = 1.148011, rho = 0.401946
> FSSEM@lambda = 0.532859, rho = 378.451849
> FSSEM@lambda = 0.532859, rho = 175.661788
> FSSEM@lambda = 0.532859, rho = 81.534979
> FSSEM@lambda = 0.532859, rho = 37.845185
> FSSEM@lambda = 0.532859, rho = 17.566179
> FSSEM@lambda = 0.532859, rho = 8.153498
> FSSEM@lambda = 0.532859, rho = 3.784518
> FSSEM@lambda = 0.532859, rho = 1.756618
> FSSEM@lambda = 0.532859, rho = 0.815350
> FSSEM@lambda = 0.532859, rho = 0.378452
> FSSEM@lambda = 0.247331, rho = 366.251814
> FSSEM@lambda = 0.247331, rho = 169.999033
> FSSEM@lambda = 0.247331, rho = 78.906561
> FSSEM@lambda = 0.247331, rho = 36.625181
> FSSEM@lambda = 0.247331, rho = 16.999903
> FSSEM@lambda = 0.247331, rho = 7.890656
> FSSEM@lambda = 0.247331, rho = 3.662518
> FSSEM@lambda = 0.247331, rho = 1.699990
> FSSEM@lambda = 0.247331, rho = 0.789066
> FSSEM@lambda = 0.247331, rho = 0.366252
> FSSEM@lambda = 0.114801, rho = 362.082773
> FSSEM@lambda = 0.114801, rho = 168.063936
> FSSEM@lambda = 0.114801, rho = 78.008369
> FSSEM@lambda = 0.114801, rho = 36.208277
> FSSEM@lambda = 0.114801, rho = 16.806394
> FSSEM@lambda = 0.114801, rho = 7.800837
> FSSEM@lambda = 0.114801, rho = 3.620828
> FSSEM@lambda = 0.114801, rho = 1.680639
```

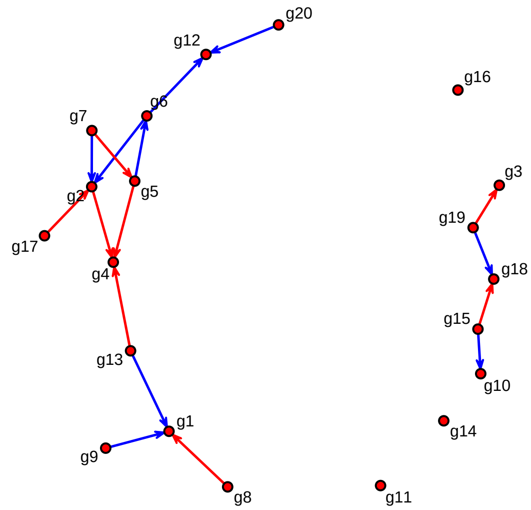

Figure 4: estimated differential GRN by fssemR

```
> FSSEM@lambda = 0.114801, rho = 0.780084
> FSSEM@lambda = 0.114801, rho = 0.362083

fit <- fitOpt$fit
```

Comparing our estimated GRNs and differential GRN with ground truth

```
cat("Power of two estimated GRNs = ",
    (TPR(fit$Bs[[1]], data$Vars$B[[1]]) + TPR(fit$Bs[[2]], data$Vars$B[[2]])) / 2)
> Power of two estimated GRNs =  1
cat("FDR of two estimated GRNs = ",
    (FDR(fit$Bs[[1]], data$Vars$B[[1]]) + FDR(fit$Bs[[2]], data$Vars$B[[2]])) / 2)
> FDR of two estimated GRNs =  0
cat("Power of estimated differential GRN = ",
    TPR(fit$Bs[[1]] - fit$Bs[[2]], data$Vars$B[[1]] - data$Vars$B[[2]]))
> Power of estimated differential GRN =  1
cat("FDR of estimated differential GRN = ",
    FDR(fit$Bs[[1]] - fit$Bs[[2]], data$Vars$B[[1]] - data$Vars$B[[2]]))
> FDR of estimated differential GRN =  0
```

From these 4 metrics, we can get the performance of our `fssemR` algorithm comparing to the ground truth (if we know)

Differential GRN Visualization

```
# data$Vars$B[[2]]    ## simulated GRN under condition 2
diffGRN = network(t(fit$Bs[[2]] - fit$Bs[[1]]) != 0, matrix.type = "adjacency", directed = TRUE)
> <sparse>[ <logic> ] : .M.sub.i.logical() maybe inefficient
# up-regulated edges are colored by `red` and down-regulated edges are colored by `blue`
ecol = 3 - sign(t(fit$Bs[[2]] - fit$Bs[[1]]))
plot(diffGRN, displaylabels = TRUE, label = network.vertex.names(GRN_2), label.cex = 0.5, edge.col = ec
```

Additionally, the differeitial effect of two GRN are also estimated. Therefore, we can tell how the interactions in two GRNs change.

```
diffGRN = Matrix::Matrix(fit$Bs[[1]] - fit$Bs[[2]], sparse = TRUE)
diffGRN
> 20 x 20 sparse Matrix of class "dgCMatrix"
>
>  [1,] .  0.0000000 . .  .         .          0.0000000 -0.201826 0.2127899
>  [2,] .  .         . .  .         0.2791508  0.3146899  0.000000 .
>  [3,] .  .         . 0  .         .          .          .        .
>  [4,] . -0.2797509 . . -0.3071442 .          .          0.000000 .
>  [5,] .  .         . .  .         .         -0.2983579  .        .
>  [6,] .  .         . .  0.2727547 .          0.0000000  .        .
>  [7,] .  .         . .  .         .          .          .        .
>  [8,] .  .         . .  .         .          .          .        .
>  [9,] .  .         . .  .         .          .          0.000000 .
> [10,] .  .         . .  .         .          .          .        .
> [11,] .  .         . .  .         0.0000000  .          0.000000 .
> [12,] .  .         . 0  .         0.3334559  .          .        .
> [13,] .  .         . .  .         .          .          .        .
> [14,] .  .         . .  .         .          .          .        .
> [15,] .  .         . .  .         .          0.0000000  .        .
> [16,] .  .         0 .  .         .          .          0.000000 .
> [17,] .  .         . .  .         .          .          .        .
> [18,] .  .         . 0  .         .          0.0000000  .        .
> [19,] .  0.0000000 . 0  .         0.0000000  0.0000000  .        .
> [20,] .  0.0000000 . .  .         .          .          .        .
>
>  [1,] . . . 0.3516458 .  .         .  .         .  0.0000000 .
>  [2,] . . . 0.0000000 0  .         . -0.3495227 .  .         .
>  [3,] . . .  .         .  .         .  .         . -0.1902061 0.0000000
>  [4,] . . . -0.3016590 .  .         .  0.0000000 .  .         .
>  [5,] . . .  .         .  .         .  .         .  .         .
>  [6,] . . . 0.0000000 .  .         .  .         .  .         .
>  [7,] . . . 0.0000000 .  .         .  .         .  .         .
>  [8,] . . .  .         .  .         .  .         .  .         .
>  [9,] . . 0  .         .  .         .  .         0  .         0.0000000
> [10,] . . .  .         .  0.3289818 .  0.0000000 .  .         .
> [11,] . . .  .         .  .         .  .         .  .         0.0000000
> [12,] . 0 .  .         0  .         .  .         .  .         0.3063937
> [13,] . . .  .         .  .         .  .         .  .         .
> [14,] . . . 0.0000000 .  .         .  .         .  .         .
> [15,] . 0 .  .         .  .         0  .         .  .         .
> [16,] . 0 .  .         .  .         .  .         .  0.0000000 0.0000000
> [17,] . . .  .         .  .         .  .         .  .         .
> [18,] 0 . 0  .         . -0.3566902 .  .         .  0.2484816 0.0000000
> [19,] . . .  .         .  .         .  .         .  .         .
> [20,] . . . 0.0000000 .  .         .  .         .  0.0000000 .
```

From the diffGRN, we can determined how the gene-gene interactions in GRN changes across two conditions, then, we can find out the key genes for condition-specific gene regulatory network.

Additionally, for more applications and the replications of our real data analysis, please go to the `https://github.com/Ivis4ml/fssemR/tree/master/inst` for more cases.

Session Information

```
sessionInfo()
> R version 3.4.0 (2017-04-21)
> Platform: x86_64-pc-linux-gnu (64-bit)
> Running under: Ubuntu 14.04.6 LTS
>
> Matrix products: default
> BLAS: /usr/lib64/microsoft-r/3.4/lib64/R/lib/libRblas.so
> LAPACK: /usr/lib64/microsoft-r/3.4/lib64/R/lib/libRlapack.so
>
> locale:
>  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
>  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
>  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
>  [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
>  [9] LC_ADDRESS=C               LC_TELEPHONE=C
> [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
>
> attached base packages:
> [1] parallel  stats     graphics  grDevices utils     datasets  methods
> [8] base
>
> other attached packages:
>  [1] Matrix_1.2-14        fssemR_0.1.4         MatrixEQTL_2.2
>  [4] sna_2.4              statnet.common_4.1.4 network_1.13.0.1
>  [7] ggnetwork_0.5.1      ggplot2_3.1.0.9000   ggnet_0.1.0
> [10] qtlnet_1.4.4         pcalg_2.4-5          graph_1.54.0
> [13] BiocGenerics_0.22.1  sem_3.1-9            igraph_1.2.2
> [16] qtl_1.44-9           RevoUtilsMath_10.0.0
>
> loaded via a namespace (and not attached):
>  [1] pkgload_1.0.2     sfsmisc_1.1-3     splines_3.4.0
>  [4] foreach_1.4.4     assertthat_0.2.0  stats4_3.4.0
>  [7] RBGL_1.52.0       yaml_2.2.0        robustbase_0.93-3
> [10] ggrepel_0.8.0     pillar_1.3.1      backports_1.1.3
> [13] lattice_0.20-35   glue_1.3.0        digest_0.6.18
> [16] ggm_2.3           minqa_1.2.4       colorspace_1.4-0
> [19] htmltools_0.3.6   fastICA_1.2-1     devtools_1.12.0
> [22] pkgconfig_2.0.2   mvtnorm_1.0-8     purrr_0.3.0
> [25] corpcor_1.6.9     scales_1.0.0.9000 lme4_1.1-20
> [28] arm_1.10-1        tibble_2.0.1      gmp_0.5-13.2
> [31] withr_2.1.2       lazyeval_0.2.1    magrittr_1.5
> [34] crayon_1.3.4      memoise_1.1.0     evaluate_0.12
> [37] nlme_3.1-137      MASS_7.3-49       xml2_1.2.0
> [40] tools_3.4.0       stringr_1.3.1     munsell_0.5.0
> [43] glmnet_2.0-16     cluster_2.0.7-1   bindrcpp_0.2.2
> [46] compiler_3.4.0    rlang_0.3.1       grid_3.4.0
> [49] nloptr_1.2.1      iterators_1.0.9   rstudioapi_0.9.0
> [52] rmarkdown_1.11    boot_1.3-20       testthat_2.0.1
> [55] mi_1.0            gtable_0.2.0      codetools_0.2-15
> [58] roxygen2_6.1.1    abind_1.4-5       R6_2.2.2
> [61] knitr_1.21        dplyr_0.7.8       bdsmatrix_1.3-3
```

```
> [64] commonmark_1.7    bindr_0.1.1       clue_0.3-56
> [67] rprojroot_1.3-2   desc_1.2.0        stringi_1.2.4
> [70] matrixcalc_1.0-3  Rcpp_1.0.0        RevoUtils_10.0.4
> [73] xfun_0.4          DEoptimR_1.0-8    tidyselect_0.2.5
> [76] coda_0.19-2
```