fssemR: Fused Sparse Structural Equation Models to Jointly Infer Gene Regulatory Networks

Xin Zhou

2022-02-11

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 `randomFSSEMdata2` 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)
>
> 'network' 1.17.1 (2021-06-12), part of the Statnet Project
> * 'news(package="network")' for changes since last version
> * 'citation("network")' for citation information
> * 'https://statnet.org' for help, support, and other information
library(ggnetwork)
> Loading required package: ggplot2
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 = 3            # 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 = 1234         # 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)
```

- 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)
```

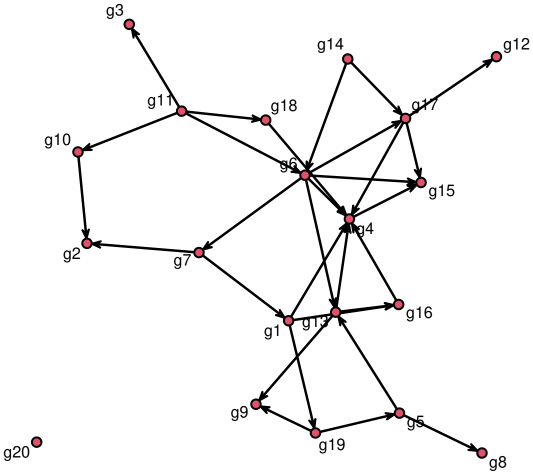

Figure 1: Simulated GRN under condition 1

```
# 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.

```
library(Matrix)
print(Matrix(data$Vars$F, sparse = TRUE))
> 20 x 60 sparse Matrix of class "dgCMatrix"
>    [[ suppressing 60 column names 'rs1', 'rs2', 'rs3' ... ]]
>
> g1  1 . . . . . . . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . .
```

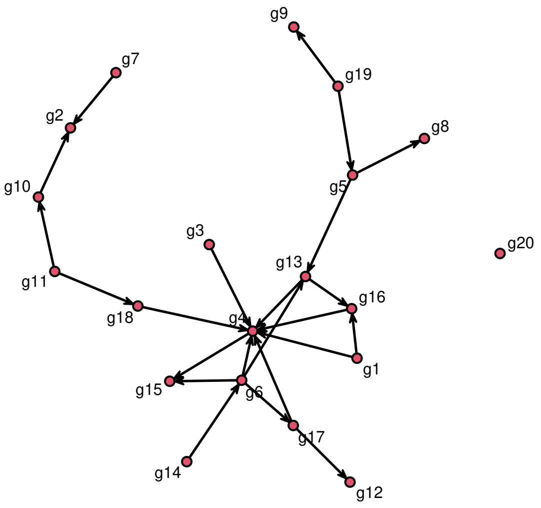

Figure 2: Simulated GRN under condition 2

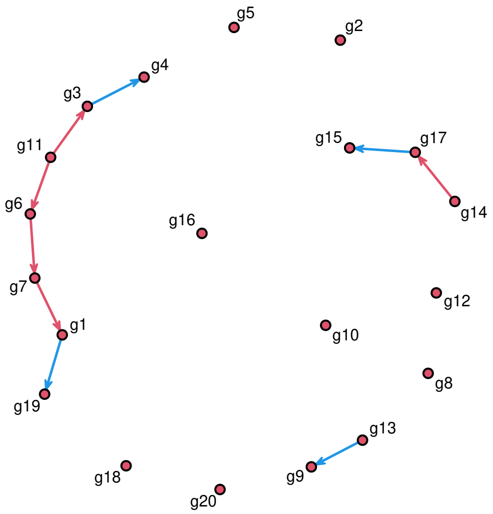

Figure 3: Simulated differential GRN (GRN2 - GRN1), up-regulated are red and down-regulated are blue

```
> 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 . . . . . . . . . . . . . . . . . . . 1 . . . . . .
> g13 . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . 1 . . . . .
> g14 . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . 1 . . . .
> g15 . . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . 1 . . .
> g16 . . . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . 1 . .
> g17 . . . . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . 1 .
> g18 . . . . . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . 1
> g19 . . . . . . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . .
> g20 . . . . . . . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . .
>
> g1  . . 1 . . . . . . . . . . . . . . . . . . .
> g2  . . . 1 . . . . . . . . . . . . . . . . . .
> g3  . . . . 1 . . . . . . . . . . . . . . . . .
> g4  . . . . . 1 . . . . . . . . . . . . . . . .
> g5  . . . . . . 1 . . . . . . . . . . . . . . .
> g6  . . . . . . . 1 . . . . . . . . . . . . . .
> g7  . . . . . . . . 1 . . . . . . . . . . . . .
> g8  . . . . . . . . . 1 . . . . . . . . . . . .
> g9  . . . . . . . . . . 1 . . . . . . . . . . .
> g10 . . . . . . . . . . . 1 . . . . . . . . . .
> g11 . . . . . . . . . . . . 1 . . . . . . . . .
> g12 . . . . . . . . . . . . . 1 . . . . . . . .
> g13 . . . . . . . . . . . . . . 1 . . . . . . .
> g14 . . . . . . . . . . . . . . . 1 . . . . . .
> g15 . . . . . . . . . . . . . . . . 1 . . . . .
> g16 . . . . . . . . . . . . . . . . . 1 . . . .
> g17 . . . . . . . . . . . . . . . . . . 1 . . .
> g18 . . . . . . . . . . . . . . . . . . . 1 . .
> g19 1 . . . . . . . . . . . . . . . . . . . 1 .
> g20 . 1 . . . . . . . . . . . . . . . . . . . 1
```

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]     [,7]
> g1  3.6002674 3.7392869 3.699864 4.181590 2.745170 4.150700 2.026133
> g2  4.0797141  7.6589754  7.171842  8.198844  6.837096 10.498451  9.383952
> g3  0.4742391 -0.6288112  0.307448  3.346940  1.995502  3.093795  2.311190
> g4 11.7753652 14.9624443 12.793385  7.808123 12.409217 12.246977 11.569742
> g5  6.6970344  9.3660441  6.201097  6.950953  7.695596 10.170404  7.186272
> g6 11.9571429 13.1616476  8.799183 10.039007 12.276469  9.551488 11.646848
>         [,8]       [,9]     [,10]     [,11]     [,12]     [,13]      [,14]
> g1  3.327513  4.9972860  1.662238  4.044778  5.273829  4.317756  3.2509141
> g2  7.072729  9.3152362  7.539168  7.147204  6.348432 10.778849  7.8427471
> g3  1.616748  0.1886544  1.952965  1.431580  3.744387  5.151211 -0.6379489
> g4 11.194027  9.8644952  9.923411 10.815945  9.430380  8.498115  9.7822356
> g5  7.954920  9.9710535  7.462603  6.213874  6.926811  9.211393  6.1870364
> g6 11.971823 11.3455078 10.211087 10.737584 10.458374  9.453512  8.8780318
>        [,15]      [,16]     [,17]      [,18]     [,19]      [,20]    [,21]
> g1  3.385443  0.7109183 0.7057026  2.8971595  3.745051  1.8656353 1.719704
> g2  7.466488  9.3958656 5.1019901  4.9209799  6.425203  4.4403237 7.682772
> g3  1.203320 -0.5115543 0.1053487 -0.0708903  3.194428  0.1938056 1.485481
> g4 10.523306 11.0447202 9.0495714  6.9280644 11.917423 12.7969411 9.487006
> g5  8.328922  8.3635613 7.2783904  8.4488568 10.336885  8.5315952 8.701464
> g6  8.025579  8.9749463 9.6546041  9.6128583 11.561384 10.7250595 9.753768
>        [,22]    [,23]     [,24]     [,25]     [,26]    [,27]      [,28]
> g1  1.554851 1.846657  3.910584  3.258208  5.188149 3.109917  4.6802711
> g2  7.371240 7.914940  4.841768  6.543653  6.999794 6.931161  5.1483194
> g3  2.380953 1.178341  1.321256  1.570854  0.562366 2.698773  0.9791985
> g4 11.598914 9.233131 10.184537 10.625274 11.230962 9.915623 13.3834346
> g5  6.280450 8.507493  7.489671  7.026706  7.511198 7.445031 10.1186721
> g6 12.395674 8.124994  8.773559 10.819460 10.016082 9.118907 10.1715932
>        [,29]     [,30]     [,31]     [,32]     [,33]     [,34]     [,35]
> g1  4.754668  3.090096  4.135533  1.595934  2.132991  3.001461  4.092259
> g2  5.118022  6.018666  5.635164  7.119504  6.433723  7.445580  9.241763
> g3  2.872081  0.439043 -1.187183  2.815723  3.280181  2.671805  3.391586
> g4 11.688971  9.503916 13.448435 10.344555  8.128460  7.820050 11.569954
> g5  7.329889  8.992139  7.235079  8.137340  8.108531  7.589989  7.375777
> g6  9.242231 10.281933 10.567314 10.391849 11.818521 10.387659  9.863558
>        [,36]    [,37]    [,38]      [,39]      [,40]    [,41]     [,42]
> g1  3.393055 5.948593 3.366081  4.0597360 -0.3663429 2.338544  3.694136
> g2  6.327874 8.364334 5.204640  8.1846127  7.7459032 6.383438  5.239194
> g3  2.984280 3.435731 2.920979  0.9263276  1.0336120 3.108278  2.385396
> g4  7.895381 8.400821 9.185198 10.1249728 10.4532589 9.134875 10.378396
> g5  7.292269 6.839445 8.013423  7.8744162  7.9953367 6.645476  6.088957
> g6 10.470441 9.914840 6.778105 11.0504625  8.9499619 8.646529  9.954485
>       [,43]     [,44]     [,45]     [,46]     [,47]     [,48]     [,49]
> g1 4.695103  2.714819  3.487214  3.020423  2.392803  3.790979  1.324226
> g2 7.584305  6.947903  9.626097  8.405025  8.362622  6.782871  7.113887
> g3 1.545335  3.601998  2.261845  2.476263  3.079334  1.298951  3.371531
> g4 9.982021  8.068697  8.700223 12.935174 11.580683 10.071533  7.989760
> g5 8.009818  7.490805  7.404173  9.433676  8.462068  8.534540  8.302042
> g6 7.783506 10.111506 11.540970 11.518222 10.475781 10.837499 10.148070
>        [,50]    [,51]    [,52]     [,53]      [,54]      [,55]     [,56]
> g1 1.6272843 4.591351 3.353593  1.845223  3.1779302  3.1959546 2.0262322
> g2 6.2114748 7.394891 8.145806  8.125161  8.4550589  7.4858709 7.1267023
> g3 0.8982972 1.442206 4.370689  1.882653  0.6298811  0.8508681 0.4498792
> g4 8.7197718 9.564964 9.437467  9.509679 11.2516733 11.1662228 6.9252817
> g5 5.9161875 8.292983 6.129373 10.058268  9.1372621  7.1645889 9.3895406
> g6 9.2361205 9.937031 9.706072  8.467976  8.9627071 10.9853081 9.4699015
>         [,57]      [,58]     [,59]     [,60]     [,61]     [,62]      [,63]
> g1  3.2534421  4.9256287  5.080000  4.757470  3.726244  2.524285  1.8379432
> g2  7.5833138  7.5391332  8.008694  8.237467  8.279457  8.517245  6.8370656
> g3  0.5385966  0.3570093 -0.411785  3.584503  2.897600  1.365270  0.6352625
> g4 12.2980293 12.2302197 10.375475 10.915019 12.671635  9.752247  9.2517521
> g5 10.0827266  8.7893001  9.147068  5.882936  7.933052  9.243591 10.3688775
> g6 10.3589433  9.6648258  7.984301  9.005356 12.348950 11.645714 11.9669184
>         [,64]    [,65]      [,66]      [,67]     [,68]     [,69]     [,70]
> g1  2.9414992 1.874796  2.6465261  0.2904542  3.889223  4.849746  1.158967
> g2  6.8770944 6.300695  7.5928195  3.2891810  5.056146  6.681896  6.248095
> g3  0.6014277 1.831094 -0.1965213  2.8128560  2.536227  1.835670  1.208708
> g4 11.3521124 6.046674  7.9838671  9.7391849  9.816755 10.018683 11.340750
> g5  8.3362385 7.108581  8.5575725 11.0567469  6.940937  8.386760  6.316186
> g6 10.4767296 7.570233 10.6822176  9.1636376 10.380589 10.677803 10.986720
>         [,71]      [,72]    [,73]     [,74]     [,75]     [,76]     [,77]
> g1  0.8480621  1.8083562 4.855467  3.102102  2.829246  3.768551  3.118184
> g2  9.0358791  6.0773554 4.451138  6.752285  5.650224  7.005774  8.955743
> g3  3.6497984  0.3554217 1.033370  1.355756  1.363667  2.714276  2.068084
> g4 10.7599105 11.4639199 9.957556 14.402776 12.861958 11.008525  9.541903
> g5  9.1918444  9.5955451 8.414815 10.388392 10.717158  5.957581  8.695705
> g6  9.8051406  9.5579511 8.212243 11.546395 10.679405 10.325284 10.544578
>         [,78]     [,79]     [,80]      [,81]      [,82]      [,83]    [,84]
> g1  0.8636715  1.406489  5.253207  2.4542662 -0.1638982  0.8439306 3.285203
> g2  7.1542059  5.486498  7.328306  4.5416536  5.8124195  6.4880217 8.485398
> g3  2.8785561  3.027524 -0.710961  0.2447047  0.8234242  1.4594185 2.661632
> g4  7.7598799 10.448922 13.736866 10.5650620  7.6348997  7.0044641 9.269359
> g5  6.7112328  9.325067  8.191811  7.7536617  7.6489689  9.4198207 8.366760
> g6 10.2651263  9.978052  9.093279  8.6144543 11.7744637 10.0131581 9.469186
>          [,85]      [,86]     [,87]    [,88]      [,89]      [,90]     [,91]
> g1  2.41441769  5.2644776  3.927249 1.753101  3.2791614  0.7199237  2.917869
> g2  7.17624650  7.1617597  7.542582 8.471949  4.3526738  7.2592582  5.679120
> g3 -0.02696729 -0.5767456  3.532073 1.949437 -0.3246674  2.2205287  1.219890
> g4 11.20803220  9.5228371  9.039677 9.380456 10.4954993  9.9334158 11.591312
> g5  8.39771546  6.7420526 10.029285 7.738503  7.2873688  6.3106563  7.835757
> g6  9.34629015 10.8304622  9.762679 9.722536  8.2257291 10.3988781 10.067661
>        [,92]     [,93]     [,94]     [,95]      [,96]      [,97]     [,98]
> g1 0.8791020  2.363968  4.423989  1.881398  3.6142524  0.1004501  6.216678
> g2 7.9076473  7.179143  7.548641  6.073591  6.9155721  7.1726482  5.953907
> g3 0.9537464  2.243399  1.937813  1.930592 -0.6493929  1.6535402  2.163758
> g4 7.9795916 10.110536 11.360789  9.036104 11.2863057  8.4139119  9.897568
> g5 8.1753283  7.335386  7.914059  7.405532  7.2843583  7.5541973  6.857763
> g6 7.4449359  9.512673 11.676398 10.489935 10.3698361 10.2434707 10.184735
>        [,99]     [,100]
> g1  5.096004  3.8474782
> g2  7.045861  6.5914387
> g3 -1.613548 -0.4581672
> g4 12.043247  9.8308726
> g5  8.107979  6.9442904
> g6  9.683423  8.0092160
head(data$Data$Y[[2]])
>         [,1]      [,2]     [,3]       [,4]     [,5]      [,6]       [,7]
> g1  8.690108  7.650089 7.638605  8.5169617 6.729960 8.5469393  8.6692288
> g2  2.995249  6.249272 2.259897  6.7458813 7.174597 5.4997007  5.2959328
> g3 10.356203 10.364683 8.586956 10.3325981 9.668369 9.4924567  7.3567339
> g4  3.549570  4.293333 5.684685  3.6354795 3.845618 5.0526936  5.6475451
> g5  2.655549  7.442200 6.322056  6.1142616 5.372684 6.2486967  5.4829668
> g6  2.851800  1.997379 3.933985  0.8675499 2.457095 0.9983611 -0.9172915
>        [,8]     [,9]     [,10]     [,11]    [,12]    [,13]    [,14]     [,15]
> g1 9.633636 7.675442 6.6682968 9.7122816 9.641088 8.670606 8.602827  9.844656
> g2 6.955494 4.359844 6.6347656 5.8819015 5.233343 5.711630 5.022570  4.797551
> g3 7.369922 8.557292 8.4478087 7.3885559 7.390418 6.409660 7.688843 10.390535
> g4 4.871609 4.358904 2.4942215 4.2474113 5.607327 4.827083 3.421167  5.999867
> g5 5.050703 5.689972 5.6270174 4.3414852 4.229479 7.031241 5.580021  5.029067
> g6 2.005659 1.272617 0.8853206 0.6181618 3.722885 2.917369 2.011497  1.233854
>       [,16]    [,17]     [,18]    [,19]      [,20]    [,21]    [,22]    [,23]
> g1 7.472449 8.734745 9.6340148 8.490372 10.6071313 9.623381 9.636093 8.637319
> g2 7.225530 6.134664 6.0950794 5.427263  5.3794125 6.199108 4.824481 3.688777
> g3 6.584107 8.316971 9.4847405 6.529197  7.3100057 8.580777 8.514300 8.548761
> g4 2.652512 3.507617 4.0780223 4.800207  4.5969690 4.174946 6.612696 2.129382
> g5 4.514112 4.278228 6.6079320 4.055462  4.3316767 4.216184 5.225491 4.945786
> g6 1.115074 1.816955 0.7404173 5.044484  0.9085611 2.680685 3.926734 0.894883
>       [,24]    [,25]    [,26]      [,27]    [,28]     [,29]     [,30]    [,31]
> g1 7.655935 6.428815 8.657636  9.5849072 8.749605 10.518635 8.6249013 9.558443
> g2 5.676344 4.653704 4.777495  5.0119238 7.364935  4.512074 5.2434420 5.231475
> g3 9.458196 7.507872 7.307869  8.2262686 7.506675  7.501701 6.2383509 9.542754
> g4 3.123536 4.088007 5.341607  4.1858725 5.719387  1.908025 4.1915981 5.362583
> g5 5.278294 4.974014 6.328720  5.5284535 7.443079  2.502859 5.1825468 7.081596
> g6 2.972932 2.367773 3.969944 -0.2436906 2.566640  1.451603 0.5835648 4.095843
>       [,32]    [,33]      [,34]     [,35]     [,36]     [,37]    [,38]    [,39]
> g1 8.554415 9.468419 8.77285107  8.435435  6.441793  8.589460 8.762908 8.679990
> g2 5.958073 6.204166 5.31963002  6.191964  6.032497  5.372422 5.291250 4.422773
> g3 9.302991 8.399147 8.63366039  9.389553  6.432201 10.500175 9.384244 6.341996
> g4 3.254670 1.705167 4.16133689  2.030891  2.208862  2.891328 2.865114 3.303643
> g5 4.972735 3.360542 5.05267883  3.654932  5.988630  5.987500 4.776129 4.942791
> g6 1.571220 3.170941 0.03828613 -1.619410 -0.309757  3.229483 2.211170 1.328322
>       [,40]    [,41]     [,42]     [,43]    [,44]    [,45]     [,46]      [,47]
> g1 8.654804 7.734561 10.643765 6.7229089 8.699734 8.607034 10.680140  7.7134910
> g2 2.229372 5.532275  4.736662 6.3628279 7.344360 2.698610  3.285532  4.5903058
> g3 8.431147 8.487411  6.553954 6.3874495 9.429893 8.562241  8.447883 10.3857080
> g4 3.596191 1.954924  7.694996 5.5617210 3.271604 3.489993  4.023961  4.1145330
> g5 4.029186 5.542300  6.080006 5.0404341 3.607566 8.007759  4.239518  6.2158029
> g6 2.396909 1.960155  1.235949 0.6603508 1.552229 1.844969  2.146796  0.9047511
>       [,48]     [,49]      [,50]     [,51]     [,52]    [,53]    [,54]    [,55]
> g1 7.442093  8.645999  7.4784015 6.5734412  9.621416 8.649487 6.595752 7.765818
> g2 4.895526  5.557559  5.7520533 6.9150500  4.182763 8.630638 4.394610 6.168345
> g3 7.427131 10.537623  8.3447434 9.5544500 10.402770 9.451298 9.374848 7.414728
> g4 6.131690  2.970880  2.8420768 1.8821962  2.793784 1.124423 2.685940 5.530576
> g5 6.700272  5.266854  6.4085534 3.7285228  2.808771 6.019985 5.903302 7.084874
> g6 2.222863  0.534237 -0.2656525 0.9803506  4.056189 1.759696 1.840698 1.506034
>       [,56]    [,57]    [,58]    [,59]    [,60]    [,61]    [,62]    [,63]
> g1 6.775749 7.613642 8.562932 8.581129 8.671421 7.606224 8.713836 8.560975
> g2 5.035944 5.248852 1.313693 8.087473 5.641239 4.857218 5.445000 4.006849
> g3 9.353864 8.532855 7.329671 9.401884 6.442289 8.436513 7.361418 8.652870
> g4 3.010573 1.860386 5.497969 3.503004 3.136619 3.949859 4.125804 3.308913
> g5 4.299090 5.468225 3.829061 6.467921 6.285294 4.904711 5.317815 5.192134
> g6 3.349186 3.119647 4.108341 1.556069 1.998411 1.246877 2.068715 1.753457
>       [,64]    [,65]    [,66]    [,67]     [,68]    [,69]    [,70]     [,71]
> g1 7.687111 6.862593 9.570485 7.722543  8.610789 8.626043 9.640670 7.5254769
> g2 5.368442 4.828919 5.044311 7.339160  5.647209 4.113753 4.120150 2.3818830
> g3 8.335705 6.470757 9.547907 8.451669 10.424976 6.454417 9.354811 6.3757493
> g4 4.204108 5.974390 4.205082 4.054356  4.195188 5.923984 2.001815 4.3207369
> g5 7.482902 5.186340 6.389785 6.676316  5.880590 5.215481 4.729986 6.4284774
> g6 3.023235 4.577137 2.845512 3.733799  2.320064 2.678657 1.028571 0.7948666
>       [,72]    [,73]    [,74]    [,75]     [,76]    [,77]    [,78]    [,79]
> g1 9.615175 7.655851 7.695215 9.740467 10.496100 6.678158 7.529596 8.682015
> g2 4.592183 5.717661 4.150367 5.012559  7.742201 4.043345 3.420831 5.556489
> g3 9.581871 5.258283 8.494385 9.397858  8.443654 9.415922 9.470876 7.428606
> g4 3.718744 3.892316 4.614173 4.138505  8.039781 3.532942 4.969553 3.666071
> g5 3.918659 4.086156 4.058949 6.214744  5.785964 6.438864 4.148291 4.438464
> g6 2.922849 1.799949 1.646551 4.915334  1.220174 1.777103 4.286966 1.679682
>       [,80]    [,81]    [,82]    [,83]     [,84]    [,85]     [,86]    [,87]
> g1 8.473960 8.654182 8.625351 9.463256 10.458962 8.852471 10.674671 9.633151
> g2 5.139325 5.289030 7.838163 4.279256  6.417873 4.762281  4.131708 6.411081
> g3 9.351422 9.574937 7.507063 6.371790  7.540650 9.374824  8.454253 9.491332
> g4 4.254202 2.642616 3.713541 3.934992  5.993575 3.277826  6.513523 5.329489
> g5 4.455718 5.984001 5.085473 3.972676  6.317460 4.094632  5.130266 5.102481
> g6 2.130944 2.785500 2.137980 2.052047  2.451524 2.176131  2.920775 4.138457
>       [,88]     [,89]     [,90]     [,91]    [,92]     [,93]    [,94]
> g1 9.684278 5.6496455 6.6620440  7.711113 7.677628  9.454476 8.568486
> g2 4.425159 4.1332174 4.2962517  2.544182 4.374363  5.245517 7.053858
> g3 8.362108 7.4877640 8.5302850 10.508835 9.430464 10.340588 9.472173
> g4 1.587472 0.9784863 3.8745224  2.853457 2.638877  5.181307 3.821083
> g5 4.913544 4.3624086 4.3810766  4.298719 7.247865  6.752673 7.553942
> g6 3.722945 1.8913453 0.7470789  3.646216 1.990653  4.723804 1.995385
>           [,95]    [,96]    [,97]    [,98]    [,99]    [,100]
> g1  7.38814235 7.679921 8.591478 9.716400 9.701708 10.440157
> g2  5.36680255 4.401467 5.958300 8.085448 5.981432  8.195674
> g3  8.47866783 7.623395 6.434612 9.557007 9.370622  6.330000
> g4  2.43786228 1.858649 5.737271 3.797275 3.568947  6.562879
> g5  4.37777814 4.948852 5.107290 5.115633 5.230804  5.534911
> g6 -0.05879901 1.287273 3.565495 2.752579 2.072428  1.878651
```

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] [,14]
> rs1    1    2    2    1    0    2    0    1    2     0     1     2     1     2
> rs2    0    1    1    1    1    2    1    1    2     1     2     2     2     1
> rs3    2    1    0    2    0    2    0    1    0     2     2     2     2     1
> rs4    0    1    2    0    1    0    2    2    1     1     0     0     2     1
> rs5    0    1    1    1    2    2    1    0    1     0     0     1     1     1
> rs6    2    2    0    2    2    1    2    2    1     1     0     1     1     1
>     [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25] [,26]
> rs1     1     1     0     2     1     2     1     1     0     1     1     2
> rs2     2     2     0     1     1     0     1     2     2     0     1     0
> rs3     1     0     1     0     1     0     2     1     0     2     0     0
> rs4     2     1     0     0     1     2     1     2     0     1     2     1
> rs5     2     1     1     1     2     1     1     0     1     1     1     1
> rs6     0     1     0     0     1     1     1     2     0     1     1     0
>     [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36] [,37] [,38]
> rs1     2     1     2     1     1     1     1     2     1     2     2     0
> rs2     1     1     0     1     0     2     1     1     2     1     1     2
> rs3     2     1     1     1     0     1     1     2     1     1     1     1
> rs4     1     2     1     0     1     2     1     0     2     0     1     2
> rs5     1     1     1     2     1     1     0     1     0     1     0     2
> rs6     0     1     1     1     1     2     1     0     2     0     2     0
>     [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48] [,49] [,50]
> rs1     1     0     0     2     1     1     1     2     1     1     1     1
> rs2     2     2     0     1     1     1     1     1     1     1     1     1
> rs3     0     1     1     1     0     2     2     1     2     1     2     1
> rs4     2     2     0     1     1     0     0     2     2     0     0     0
> rs5     0     1     1     1     1     0     1     1     1     1     1     0
> rs6     2     1     1     1     0     2     1     2     1     1     1     1
>     [,51] [,52] [,53] [,54] [,55] [,56] [,57] [,58] [,59] [,60] [,61] [,62]
> rs1     2     1     0     2     1     1     2     2     2     1     1     1
> rs2     2     2     2     2     1     1     2     1     2     2     2     1
> rs3     0     2     2     2     0     1     0     1     0     2     1     1
> rs4     0     0     1     2     2     0     1     2     1     1     1     0
> rs5     2     0     2     2     1     2     2     0     2     0     1     1
> rs6     1     0     0     1     2     1     0     1     1     1     1     1
>     [,63] [,64] [,65] [,66] [,67] [,68] [,69] [,70] [,71] [,72] [,73] [,74]
> rs1     1     1     1     0     0     1     2     1     0     1     2     0
> rs2     1     1     0     1     1     1     1     0     2     0     1     2
> rs3     0     0     0     0     2     2     2     1     2     1     1     1
> rs4     1     2     1     1     1     1     1     2     1     2     1     2
> rs5     1     1     1     1     2     0     2     0     2     1     1     1
> rs6     2     1     0     1     0     2     1     1     1     2     1     2
>     [,75] [,76] [,77] [,78] [,79] [,80] [,81] [,82] [,83] [,84] [,85] [,86]
> rs1     0     1     1     0     0     1     1     0     0     1     1     1
> rs2     0     0     1     2     1     1     1     0     1     1     1     1
> rs3     1     1     1     2     2     0     2     1     1     1     0     0
> rs4     2     2     0     0     1     2     1     1     0     1     2     0
> rs5     2     0     2     1     1     1     0     0     1     1     1     0
> rs6     2     0     0     1     1     0     1     1     1     0     1     1
>     [,87] [,88] [,89] [,90] [,91] [,92] [,93] [,94] [,95] [,96] [,97] [,98]
> rs1     2     0     2     0     0     0     2     1     0     1     0     2
> rs2     1     2     1     2     0     1     2     1     2     1     1     1
> rs3     2     1     1     0     2     0     2     2     0     0     2     1
> rs4     1     1     1     2     2     1     1     1     0     1     1     1
> rs5     2     1     0     0     1     2     1     2     0     0     1     1
> rs6     1     1     1     1     1     0     1     1     2     1     1     1
>     [,99] [,100]
> rs1     2      2
> rs2     1      1
> rs3     0      1
> rs4     1      1
> rs5     2      1
> rs6     1      0
head(data$Data$X[[2]] - 1)
>     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13] [,14]
> rs1    1    0    1    0    0    0    1    1    1     0     2     2     1     2
> rs2    1    1    0    1    1    1    1    2    1     2     2     0     1     0
> rs3    1    2    2    1    2    1    1    2    2     1     0     1     0     1
> rs4    2    1    2    1    2    2    2    1    0     1     2     2     0     1
> rs5    1    2    1    1    1    1    0    1    2     1     1     0     1     1
> rs6    1    1    1    1    1    0    0    2    0     1     0     1     1     1
>     [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25] [,26]
> rs1     1     1     0     1     0     1     1     2     1     0     0     1
> rs2     2     0     0     1     1     1     1     0     0     2     0     1
> rs3     2     0     1     2     0     1     1     2     2     1     0     0
> rs4     2     1     1     2     1     1     1     2     1     1     1     1
> rs5     1     0     1     2     0     1     1     2     1     1     1     1
> rs6     1     0     1     1     2     1     2     2     0     1     2     2
>     [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36] [,37] [,38]
> rs1     2     1     2     1     1     0     2     1     0     0     2     2
> rs2     1     1     2     1     2     2     2     2     2     2     1     0
> rs3     1     1     0     0     1     2     0     1     2     1     2     1
> rs4     2     2     1     2     1     2     0     0     0     1     1     2
> rs5     1     1     0     2     1     1     0     0     0     2     2     1
> rs6     0     0     1     0     2     1     2     0     0     0     1     1
>     [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48] [,49] [,50]
> rs1     1     1     0     2     0     1     2     2     1     1     0     1
> rs2     1     0     1     1     1     2     0     1     1     1     1     0
> rs3     0     1     1     0     0     1     1     1     2     1     2     1
> rs4     0     1     1     1     2     0     1     1     2     1     2     1
> rs5     1     0     2     2     1     1     2     0     1     2     1     2
> rs6     1     1     1     1     1     0     1     1     2     2     0     0
>     [,51] [,52] [,53] [,54] [,55] [,56] [,57] [,58] [,59] [,60] [,61] [,62]
> rs1     0     2     1     1     2     1     0     1     2     0     2     1
> rs2     1     1     1     2     1     1     1     0     2     1     1     2
> rs3     2     2     1     2     2     1     1     2     2     0     1     0
> rs4     2     0     0     0     1     0     0     1     1     0     2     1
> rs5     0     1     2     1     2     1     0     0     1     2     2     0
> rs6     0     1     0     2     1     1     0     1     2     1     0     1
>     [,63] [,64] [,65] [,66] [,67] [,68] [,69] [,70] [,71] [,72] [,73] [,74]
> rs1     1     0     1     2     1     1     2     1     1     2     0     1
> rs2     1     1     1     1     2     1     0     1     0     1     1     1
> rs3     1     1     0     2     1     2     0     1     0     2     0     2
> rs4     1     1     1     0     0     1     1     0     1     0     1     1
> rs5     2     1     1     1     1     1     1     1     1     1     0     0
> rs6     1     2     2     2     2     0     1     1     1     1     2     0
>     [,75] [,76] [,77] [,78] [,79] [,80] [,81] [,82] [,83] [,84] [,85] [,86]
> rs1     2     2     0     1     2     0     1     1     2     2     0     2
> rs2     0     1     1     1     1     1     1     1     2     1     2     2
> rs3     2     1     1     2     0     1     2     0     1     1     1     2
> rs4     1     2     2     1     1     1     1     1     1     2     1     2
> rs5     2     1     1     1     1     0     2     1     1     2     1     1
> rs6     2     1     1     1     1     0     2     1     0     0     1     2
>     [,87] [,88] [,89] [,90] [,91] [,92] [,93] [,94] [,95] [,96] [,97] [,98]
> rs1     1     1     0     0     2     0     1     1     1     0     1     2
> rs2     1     0     1     0     1     0     2     1     1     1     2     2
> rs3     1     2     2     1     1     2     2     2     1     1     0     2
> rs4     1     0     0     1     1     1     1     1     1     1     2     1
> rs5     0     1     1     1     1     1     2     1     1     2     1     1
> rs6     2     2     1     0     2     0     2     1     0     1     2     2
>     [,99] [,100]
> rs1     1      1
> rs2     1      2
> rs3     1      1
> rs4     1      1
> rs5     1      1
> rs6     1      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] 12.023141 11.918540 11.796338 11.654364 11.490473 11.302632 11.089054
>  [8] 10.848336 10.579608 10.282677  9.958135  9.607442  9.232943  8.837852
> [15]  8.426159  8.002514  7.572067  7.140305  6.712875  6.295423  5.893438
> [22]  5.512094  5.156088  4.829472  4.535450  4.276188  4.052645  3.864478
> [29]  3.710059  3.586640  3.490644  3.418029  3.364653  3.326590  3.300337
> [36]  3.282926  3.271949  3.265519  3.262211  3.260970  3.261043  3.261901
> [43]  3.263185  3.264656  3.266165  3.267618  3.268964  3.270181  3.271257
> [50]  3.272197
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 = 148.644902, rho = 0.000000
> FSSEM@lambda = 148.644902, rho = 0.000000
> FSSEM@lambda = 148.644902, rho = 0.000000
> FSSEM@lambda = 148.644902, rho = 0.000000
> FSSEM@lambda = 148.644902, rho = 0.000000
> FSSEM@lambda = 148.644902, rho = 0.000000
> FSSEM@lambda = 148.644902, rho = 0.000000
> FSSEM@lambda = 148.644902, rho = 0.000000
> FSSEM@lambda = 148.644902, rho = 0.000000
> FSSEM@lambda = 148.644902, rho = 0.000000
> FSSEM@lambda = 68.994852, rho = 4.131694
> FSSEM@lambda = 68.994852, rho = 1.917763
> FSSEM@lambda = 68.994852, rho = 0.890147
> FSSEM@lambda = 68.994852, rho = 0.413169
> FSSEM@lambda = 68.994852, rho = 0.191776
> FSSEM@lambda = 68.994852, rho = 0.089015
> FSSEM@lambda = 68.994852, rho = 0.041317
> FSSEM@lambda = 68.994852, rho = 0.019178
> FSSEM@lambda = 68.994852, rho = 0.008901
> FSSEM@lambda = 68.994852, rho = 0.004132
> FSSEM@lambda = 32.024573, rho = 4.732226
> FSSEM@lambda = 32.024573, rho = 2.196505
> FSSEM@lambda = 32.024573, rho = 1.019527
> FSSEM@lambda = 32.024573, rho = 0.473223
> FSSEM@lambda = 32.024573, rho = 0.219650
> FSSEM@lambda = 32.024573, rho = 0.101953
> FSSEM@lambda = 32.024573, rho = 0.047322
> FSSEM@lambda = 32.024573, rho = 0.021965
> FSSEM@lambda = 32.024573, rho = 0.010195
> FSSEM@lambda = 32.024573, rho = 0.004732
> FSSEM@lambda = 14.864490, rho = 4.921283
> FSSEM@lambda = 14.864490, rho = 2.284257
> FSSEM@lambda = 14.864490, rho = 1.060258
> FSSEM@lambda = 14.864490, rho = 0.492128
> FSSEM@lambda = 14.864490, rho = 0.228426
> FSSEM@lambda = 14.864490, rho = 0.106026
> FSSEM@lambda = 14.864490, rho = 0.049213
> FSSEM@lambda = 14.864490, rho = 0.022843
> FSSEM@lambda = 14.864490, rho = 0.010603
> FSSEM@lambda = 14.864490, rho = 0.004921
> FSSEM@lambda = 6.899485, rho = 364.405179
> FSSEM@lambda = 6.899485, rho = 169.141901
> FSSEM@lambda = 6.899485, rho = 78.508716
> FSSEM@lambda = 6.899485, rho = 36.440518
> FSSEM@lambda = 6.899485, rho = 16.914190
> FSSEM@lambda = 6.899485, rho = 7.850872
> FSSEM@lambda = 6.899485, rho = 3.644052
> FSSEM@lambda = 6.899485, rho = 1.691419
> FSSEM@lambda = 6.899485, rho = 0.785087
> FSSEM@lambda = 6.899485, rho = 0.364405
> FSSEM@lambda = 3.202457, rho = 532.252704
> FSSEM@lambda = 3.202457, rho = 247.049821
> FSSEM@lambda = 3.202457, rho = 114.670369
> FSSEM@lambda = 3.202457, rho = 53.225270
> FSSEM@lambda = 3.202457, rho = 24.704982
> FSSEM@lambda = 3.202457, rho = 11.467037
> FSSEM@lambda = 3.202457, rho = 5.322527
> FSSEM@lambda = 3.202457, rho = 2.470498
> FSSEM@lambda = 3.202457, rho = 1.146704
> FSSEM@lambda = 3.202457, rho = 0.532253
> FSSEM@lambda = 1.486449, rho = 486.678491
> FSSEM@lambda = 1.486449, rho = 225.896145
> FSSEM@lambda = 1.486449, rho = 104.851702
> FSSEM@lambda = 1.486449, rho = 48.667849
> FSSEM@lambda = 1.486449, rho = 22.589615
> FSSEM@lambda = 1.486449, rho = 10.485170
> FSSEM@lambda = 1.486449, rho = 4.866785
> FSSEM@lambda = 1.486449, rho = 2.258961
> FSSEM@lambda = 1.486449, rho = 1.048517
> FSSEM@lambda = 1.486449, rho = 0.486678
> FSSEM@lambda = 0.689949, rho = 454.769361
> FSSEM@lambda = 0.689949, rho = 211.085239
> FSSEM@lambda = 0.689949, rho = 97.977089
> FSSEM@lambda = 0.689949, rho = 45.476936
> FSSEM@lambda = 0.689949, rho = 21.108524
> FSSEM@lambda = 0.689949, rho = 9.797709
> FSSEM@lambda = 0.689949, rho = 4.547694
> FSSEM@lambda = 0.689949, rho = 2.110852
> FSSEM@lambda = 0.689949, rho = 0.979771
> FSSEM@lambda = 0.689949, rho = 0.454769
> FSSEM@lambda = 0.320246, rho = 434.071454
> FSSEM@lambda = 0.320246, rho = 201.478121
> FSSEM@lambda = 0.320246, rho = 93.517860
> FSSEM@lambda = 0.320246, rho = 43.407145
> FSSEM@lambda = 0.320246, rho = 20.147812
> FSSEM@lambda = 0.320246, rho = 9.351786
> FSSEM@lambda = 0.320246, rho = 4.340715
> FSSEM@lambda = 0.320246, rho = 2.014781
> FSSEM@lambda = 0.320246, rho = 0.935179
> FSSEM@lambda = 0.320246, rho = 0.434071
> FSSEM@lambda = 0.148645, rho = 407.235384
> FSSEM@lambda = 0.148645, rho = 189.021921
> FSSEM@lambda = 0.148645, rho = 87.736204
> FSSEM@lambda = 0.148645, rho = 40.723538
> FSSEM@lambda = 0.148645, rho = 18.902192
> FSSEM@lambda = 0.148645, rho = 8.773620
> FSSEM@lambda = 0.148645, rho = 4.072354
> FSSEM@lambda = 0.148645, rho = 1.890219
> FSSEM@lambda = 0.148645, rho = 0.877362
> FSSEM@lambda = 0.148645, rho = 0.407235

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
```

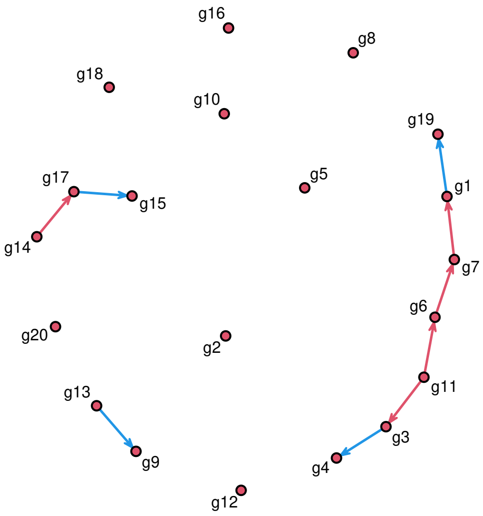

Figure 4: estimated differential GRN by fssemR

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)
rownames(diffGRN) = colnames(diffGRN) = rownames(data$Vars$B[[1]])
diffGRN
> 20 x 20 sparse Matrix of class "dgCMatrix"
>    [[ suppressing 20 column names 'g1', 'g2', 'g3' ... ]]
>
> g1  .         . .         . .  .         -0.2142346 . . .  .         .
> g2  .         . .         . .  .          0.0000000 . . 0  .         .
> g3  .         . .         . .  .          .         . . . -0.3007874 .
> g4  0.0000000 . 0.2859799 . .  0.0000000  .         . . .  .         .
> g5  .         . .         . .  .          .         . . .  .         .
> g6  .         . .         . .  .          .         . . . -0.2557159 .
> g7  .         . .         . . -0.2133412  .         . . .  .         .
> g8  .         . .         . 0  .          .         . . .  .         .
> g9  .         . .         . .  .          .         . . .  .         .
> g10 .         . .         . .  .          .         . . .  0.0000000 .
> g11 .         . .         . .  .          .         . . .  .         .
> g12 .         . .         . .  .          .         . . .  .         .
> g13 .         . .         . 0  0.0000000  .         . . .  .         .
> g14 .         . .         . .  .          .         . . .  .         .
> g15 .         . .         0 .  0.0000000  .         . . .  .         .
> g16 0.0000000 . .         . .  .          .         . . .  .         .
> g17 .         . .         . .  0.0000000  .         . . .  .         .
> g18 .         . .         . .  .          .         . . .  0.0000000 .
> g19 0.2313426 . .         . .  .          .         . . .  .         .
> g20 .         . .         . .  .          .         . . .  .         .
>
> g1  .          .         . . .         . . .
> g2  .          .         . . .         . . .
> g3  .          .         . . .         . . .
> g4  0.0000000  .         . 0 0.0000000 0 . .
> g5  .          .         . . .         . 0 .
> g6  .          0.0000000 . . .         . . .
> g7  .          .         . . .         . . .
> g8  .          .         . . .         . . .
> g9  0.1967908  .         . . .         . 0 .
> g10 .          .         . . .         . . .
> g11 .          .         . . .         . . .
> g12 .          .         . . 0.0000000 . . .
> g13 .          .         . . .         . . .
> g14 .          .         . . .         . . .
> g15 .          .         . . 0.3366277 . . .
> g16 0.0000000  .         . . .         . . .
> g17 .         -0.2635346 . . .         . . .
> g18 .          .         . . .         . . .
> g19 .          .         . . .         . . .
> g20 .          .         . . .         . . .
```

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 4.1.2 (2021-11-01)
> Platform: x86_64-pc-linux-gnu (64-bit)
> Running under: Ubuntu 20.04.3 LTS
> 
> Matrix products: default
> BLAS:   /usr/lib/x86_64-linux-gnu/openblas-pthread/libblas.so.3
> LAPACK: /usr/lib/x86_64-linux-gnu/openblas-pthread/liblapack.so.3
> 
> locale:
>  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C              
>  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=C              
>  [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] stats     graphics  grDevices utils     datasets  methods   base     
> 
> other attached packages:
> [1] Matrix_1.4-0     ggnetwork_0.5.10 ggplot2_3.3.5    network_1.17.1  
> [5] fssemR_0.1.8    
> 
> loaded via a namespace (and not attached):
>  [1] shape_1.4.6          tidyselect_1.1.1     xfun_0.29           
>  [4] purrr_0.3.4          splines_4.1.2        lattice_0.20-45     
>  [7] colorspace_2.0-2     vctrs_0.3.8          generics_0.1.2      
> [10] htmltools_0.5.2      yaml_2.2.2           utf8_1.2.2          
> [13] survival_3.2-13      rlang_1.0.1          pillar_1.7.0        
> [16] glue_1.6.1           withr_2.4.3          foreach_1.5.2       
> [19] lifecycle_1.0.1      stringr_1.4.0        munsell_0.5.0       
> [22] gtable_0.3.0         mvtnorm_1.1-3        codetools_0.2-18    
> [25] coda_0.19-4          evaluate_0.14        knitr_1.37          
> [28] fastmap_1.1.0        parallel_4.1.2       fansi_1.0.2         
> [31] Rcpp_1.0.8           scales_1.1.1         digest_0.6.29       
> [34] stringi_1.7.6        dplyr_1.0.8          grid_4.1.2          
> [37] cli_3.1.1            tools_4.1.2          magrittr_2.0.2      
> [40] glmnet_4.1-3         tibble_3.1.6         crayon_1.4.2        
> [43] pkgconfig_2.0.3      MASS_7.3-55          ellipsis_0.3.2      
> [46] rmarkdown_2.11       rstudioapi_0.13      iterators_1.0.14    
> [49] statnet.common_4.5.0 qtl_1.50             R6_2.5.1            
> [52] igraph_1.2.11        compiler_4.1.2
```