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

Xin Zhou

2019-12-03

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)
> network: Classes for Relational Data
> Version 1.13.0.1 created on 2015-08-31.
> copyright (c) 2005, Carter T. Butts, University of California-Irvine
>                     Mark S. Handcock, University of California -- Los Angeles
>                     David R. Hunter, Penn State University
>                     Martina Morris, University of Washington
>                     Skye Bender-deMoll, University of Washington
>  For citation information, type citation("network").
>  Type help("network-package") to get started.
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
```

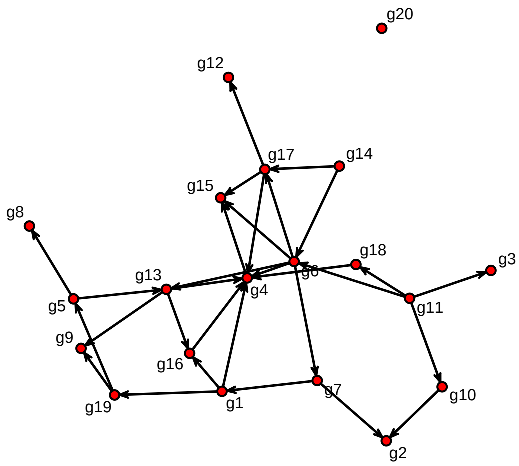

Figure 1: Simulated GRN under condition 1

```
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)
```

```
# 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' ... ]]
>
```

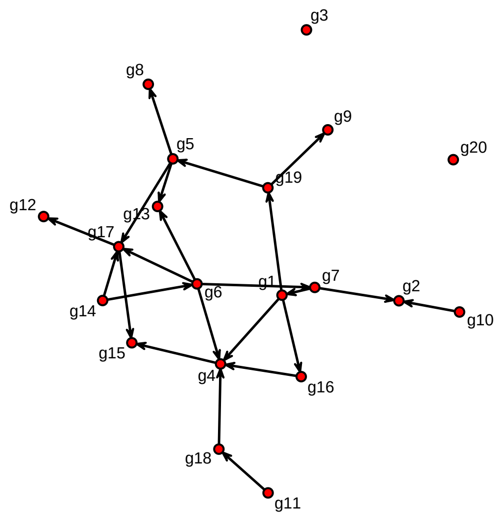

Figure 2: Simulated GRN under condition 2

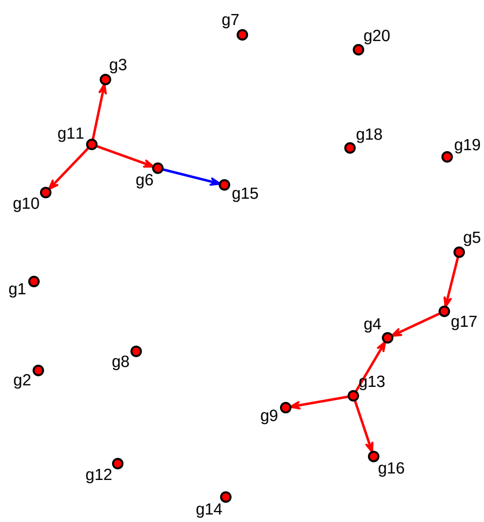

Figure 3: Simulated differential GRN (GRN2 - GRN1), up-regulated are red and down-regulated are blue

```
> 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 . . . . . . . . . . . . . . . . . . . 1 . . .
> g13 . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . 1 . .
> g14 . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . 1 .
> g15 . . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . 1
> g16 . . . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . .
> g17 . . . . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . .
> g18 . . . . . . . . . . . . . . . . . 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 . . . . . . . . . . . . . . . . . . . 1 . . . .
> g17 . 1 . . . . . . . . . . . . . . . . . . . 1 . . .
> g18 . . 1 . . . . . . . . . . . . . . . . . . . 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]
> g1   2.770448   6.259964   6.674162   6.808461   5.680147  7.318518
> g2  16.693933  16.400043  17.907393  20.717025  18.462004 19.310053
> g3  -4.143627  -1.537331  -1.487832  -6.339450  -3.506562 -3.625918
> g4 -13.823825 -11.027926 -13.736342 -10.500963 -11.822357 -7.312691
> g5   3.334527   4.691596   1.231610   1.863935   3.921532  1.092439
> g6  11.946279  12.189177  12.592806   9.531924  12.283651  9.101853
>           [,7]       [,8]      [,9]      [,10]      [,11]       [,12]
> g1   7.925731   5.469580  6.874138   6.259060   4.991997   3.6558171
> g2  20.064330  18.625910 17.263045  18.031644  17.971561  19.3634073
> g3  -2.821562  -3.989399 -4.372809  -4.848280  -3.273691  -4.9118578
> g4 -10.882462 -11.926105 -9.673294 -11.874307 -13.799269 -11.9793126
> g5   3.386369   3.528428  2.820687   1.429835   2.773986   0.9921462
> g6  10.454074   9.374013 11.899145  11.914369  11.560273   9.3255531
>          [,13]       [,14]       [,15]       [,16]       [,17]       [,18]
> g1  9.0770168   5.6315452   6.6524100   7.8541907   2.7487485   3.9887650
> g2 21.5317595  15.9797857  18.7269834  15.6236696  15.2581297  16.8721849
> g3 -3.7886454  -4.9133877  -2.9578181  -2.7955708  -4.1907567  -7.0963695
> g4 -7.1027533 -12.9192965 -10.8718204 -10.2683217 -13.0916818 -12.3372152
> g5  0.4206075   0.2349598  -0.4666022   0.5809634   0.3444712   0.2551614
> g6  8.7646739  10.0190473  12.4381136  11.0151567  11.3150153  11.5710326
>         [,19]      [,20]     [,21]      [,22]       [,23]       [,24]
> g1  2.424377   3.139723  5.882116   7.097460   6.4267704   3.5841797
> g2 19.515048  17.469934 17.616417  20.214885  18.7438415  18.9216099
> g3 -3.976164  -1.754099 -4.261824  -3.903713  -4.3920988  -3.5657696
> g4 -9.581770 -11.708551 -9.683074 -12.941502 -10.0141733 -13.5747447
> g5  3.865410   2.042421  1.629456   3.318893  -0.6147502  -0.1247361
> g6  7.782795  10.619403 11.493185  12.700910  11.0183445  11.1710466
>          [,25]       [,26]        [,27]      [,28]       [,29]      [,30]
> g1   4.817649   4.7659734   4.89861776   4.622141   2.7658565   4.349978
> g2  18.714211  16.7075823  19.73046183  17.458651  18.8136511  17.162172
> g3  -4.691829  -3.3885190  -3.84873719  -1.869941  -2.6170182  -4.288225
> g4 -11.879677 -11.5629418 -12.19956473 -10.094749 -13.1710101 -10.217128
> g5   2.274330   0.8840928   0.04969382   1.595256   0.8528509   1.651465
> g6   9.396885  10.4390922  10.72582492  12.284884  12.1070376   9.195672
>          [,31]      [,32]      [,33]      [,34]      [,35]      [,36]
> g1   3.543636   5.954406   3.907671   5.046753   7.990178   6.327497
> g2  16.878461  19.035679  20.335273  19.003801  21.034857  19.952476
> g3  -1.726678  -3.761303  -4.577775  -6.274101  -2.634423  -5.533067
> g4 -12.532556 -11.581748 -10.914734 -10.954571 -10.224966 -12.476030
> g5   2.662231   1.893884   2.981806   1.274090   1.649356   1.757932
> g6  10.596832  12.681480   9.981001   9.478938  10.343363  13.270179
>          [,37]      [,38]      [,39]      [,40]      [,41]      [,42]
> g1   5.720402   3.813243   5.946461   5.888265   3.684659   3.672092
> g2  19.863058  18.437618  16.550861  18.320714  18.209933  18.986330
> g3  -5.587944  -2.793105  -3.958480  -2.841557  -3.885837  -3.955722
> g4 -12.073591 -11.487370 -11.252530 -11.105414 -13.445681 -14.351061
> g5   1.070389  -2.169704   3.096173   2.099796  -1.013806   1.311431
> g6  10.340032  12.008636  11.363062  12.055004  10.158709  11.769262
>           [,43]      [,44]      [,45]     [,46]     [,47]      [,48]
> g1   4.5917122   4.295282   6.524264  4.906347  6.536732   4.879435
> g2  17.6616356  19.092571  16.980908 17.926787 20.854698  18.096823
```

```
> g3  -4.1945620  -5.919006  -5.307858 -2.065621 -2.911331  -4.558058
> g4 -11.3283471 -13.501180 -10.086630 -9.918077 -9.818009 -11.012461
> g5  -0.9669337   1.025179   2.193783  2.431141  2.891142   2.098039
> g6   9.9449443  12.121816   8.819851 10.539147  8.886218  11.969230
>          [,49]       [,50]      [,51]      [,52]      [,53]      [,54]
> g1    5.805099   5.3481447   6.542402   4.548018  6.5052474  5.6800691
> g2   19.893654  17.1882774  18.428548  20.609397 18.2871916 16.4169909
> g3   -5.709911  -4.9131706  -3.418575  -4.720646 -2.5454282 -3.0046541
> g4 -12.567286 -14.6149848 -10.212645 -13.427252 -8.5490332 -9.3466052
> g5    1.445014   0.7621789   1.626611   0.504638 -0.5208959  0.4608354
> g6   12.931516  12.1752790  11.583177   9.203124 11.0028094  9.1496536
>          [,55]      [,56]     [,57]      [,58]       [,59]       [,60]
> g1    4.812958  4.8661167  4.869438   5.344981  6.20694731   5.5725124
> g2   16.738751 17.9204878 18.189879  16.570016 15.91771238  19.7115401
> g3   -2.779173 -5.5457626 -2.990251  -3.306833 -2.64457257  -3.3681411
> g4 -11.626939 -9.6168655 -9.334655 -10.442561 -9.57074956 -14.3716526
> g5    2.339131  0.1455553  2.476909   1.206455 -0.01564206   0.1179841
> g6   10.106026 10.2269138  9.342599  10.693833  9.69753120  12.2839852
>          [,61]     [,62]     [,63]      [,64]      [,65]      [,66]
> g1    7.342388  6.648037  3.600893   5.336076   4.133792   5.465963
> g2   20.198763 18.900766 16.701180  17.194249  17.942103  15.649900
> g3   -3.813033 -6.186200 -5.873100  -1.879813  -5.503249  -5.839342
> g4 -10.967209 -9.121562 -9.566319 -11.432820 -13.677219 -11.592131
> g5    3.129266  3.658870  3.548002   1.136549  -1.332778   2.128129
> g6   12.412034  8.365781 10.289357  11.048246  11.253684  12.057317
>           [,67]      [,68]      [,69]      [,70]     [,71]      [,72]
> g1    3.1178414   3.040707   6.146053   4.658552  7.422017   4.866166
> g2   19.0049328  17.458261  18.793092  18.685292 20.294841  16.868347
> g3   -4.0496717  -3.996697  -4.685695  -2.634156 -3.524620  -1.687134
> g4 -10.9202425 -13.516270 -10.377952 -13.706729 -7.970190 -11.624714
> g5    0.2225315   1.331506   2.379584   2.996124  1.710600   1.058597
> g6   13.6685479  11.302242  12.325844  10.103907 10.001830  12.066131
>            [,73]      [,74]     [,75]      [,76]      [,77]      [,78]
> g1    3.58635809   5.690741  4.172261   5.387820   5.621021   5.394674
> g2   18.82645816  18.847631 18.629829  19.041418  17.659048  19.329822
> g3   -3.75584938  -1.655152 -1.811239  -4.112731  -4.263769  -5.582238
> g4 -11.47367684 -10.025294 -9.161142 -13.562384 -10.564153 -13.852200
> g5    0.01163896   3.577268  1.980380   1.032987   2.770731   1.533226
> g6   10.69955663  11.212962 12.269603  11.884114  10.636644  11.152263
>           [,79]       [,80]        [,81]      [,82]      [,83]       [,84]
> g1    4.4105779   4.6029323   4.56500496   4.367527   5.380164   5.4273765
> g2   19.1410904  15.7169809  17.53883594  17.419603  17.960849  18.3381625
> g3   -4.9166690  -0.6730186  -3.81234016  -4.264212  -6.271250  -4.0917934
> g4 -10.9541582 -10.6823470 -12.48190764 -12.755898 -10.219740 -11.0548885
> g5    0.8515518   1.0503681   0.02597338   2.699285   0.721474  -0.3265995
> g6   12.8161875   8.9421132  12.63085308  11.538503  11.722178   8.7453121
>          [,85]      [,86]     [,87]      [,88]        [,89]      [,90]
> g1    4.653092   4.744772  5.868073   6.560494   4.41374313   5.610824
> g2   17.109922  16.015486 19.842632  19.069869  17.93558000  18.103617
> g3   -2.571671  -5.659185 -4.608614  -4.458150  -4.10731537  -4.203695
> g4 -11.068923 -12.624786 -9.291319 -11.927320 -13.02321097 -12.811917
> g5    1.289552   2.282238  1.422763   1.622764   0.09502152   2.123966
> g6    9.337081  10.390021 11.648862  12.417930  12.86118757  11.585181
```

```
>          [,91]        [,92]      [,93]      [,94]      [,95]      [,96]
> g1   3.655231   5.92297709   5.594532   5.727926   3.839261   4.929862
> g2  17.851294  18.32045224  18.777437  20.073689  17.657657  16.148851
> g3  -3.380978  -5.53142388  -3.585946  -4.674287  -4.920095  -2.588720
> g4 -12.834692 -11.87266829 -12.290983 -11.395479 -12.027204 -13.453395
> g5   1.166630   0.03784201   1.726572   3.306152   1.073038   1.239465
> g6  10.954648  11.47770793  12.996580   8.922177  11.065604  11.828441
>          [,97]      [,98]      [,99]      [,100]
> g1   5.060089   2.481965   5.454253   5.0774444
> g2  19.692290  17.271134  15.537260  15.8655382
> g3  -4.892736  -4.263437  -3.641717  -3.9794759
> g4 -11.633394 -13.209137 -11.864635 -12.7489051
> g5   2.277451   1.169117   1.673870  -0.1733115
> g6   9.993153  10.906894  10.310814  10.5183231
head(data$Data$Y[[2]])
>          [,1]       [,2]         [,3]      [,4]       [,5]        [,6]
> g1  9.948509 14.1902215  9.472175410 13.525595 15.2023393 13.35551608
> g2 13.693329 14.5491579 12.236147973 13.832619 13.1579193 13.62498285
> g3  9.944288 10.8600114  9.853991067  9.957988 10.0641103 12.05572559
> g4  7.608072 14.0317228 10.999923388 12.418088 11.9441781 11.69112129
> g5 -1.001508 -1.6420794 -0.007065345 -2.260965 -1.2238051 -2.31613817
> g6 -1.504758 -0.7014416 -1.956536055 -2.055484 -0.2329952  0.06656525
>           [,7]      [,8]       [,9]      [,10]      [,11]      [,12]
> g1 12.9313841 15.920320 11.7873512 15.5284406 13.0943409 12.2718422
> g2 11.1914649 11.125735 12.5660697 12.7882782 11.0122571 10.2109819
> g3 11.8469884  9.919121  9.8297656  9.8182069 10.7495371  9.8810545
> g4 10.3481056 10.710947 10.3674912 12.1965693 10.6737369  8.6713656
> g5 -4.4396905 -1.397609 -1.9925676 -2.4034781 -3.2913450  0.2756007
> g6 -0.6787107  1.502859 -0.5370119  0.1933713 -0.7247887 -0.2251454
>          [,13]     [,14]     [,15]     [,16]      [,17]      [,18]
> g1 12.3012146 11.038065 12.847408 13.371809 13.7041637 14.5090985
> g2 10.2226867 10.562072 14.190406  9.303599 12.0570419 13.5412106
> g3  9.0253133 10.013826 12.926474  8.829644 10.0008158  9.9720776
> g4 12.6354302 10.859482 10.103639 10.648757 11.1163683 12.2015056
> g5 -0.7831814 -2.277604 -2.748106 -2.634764 -1.5775282 -3.1704298
> g6 -2.6685885 -1.469859 -1.149889 -2.490966 -0.9729241 -0.5301707
>         [,19]      [,20]      [,21]      [,22]     [,23]      [,24]
> g1 11.299426 14.0388550 14.1814221 12.8373883  9.508576 13.0064096
> g2  9.904300 11.7649206 12.7210911 12.3104639 11.122384 12.2937752
> g3  8.989161  9.9067289 11.0145523 10.8886026  8.869276  8.7895813
> g4 10.336963  9.9031883 11.4299294 11.0511164  9.582806 10.9271338
> g5  1.263874 -2.7823693 -0.5245449  0.5337292 -3.315519 -0.5860221
> g6 -3.774797  0.6910732 -1.9678048 -1.0686266 -2.635195 -1.2533016
>         [,25]      [,26]     [,27]      [,28]      [,29]     [,30]
> g1 10.732606 12.6257636 10.482928 14.1584787 12.6756832 12.942063
> g2  9.994278 10.5424698 11.205824 10.6823609 10.6087761 10.643474
> g3 10.091188  9.7648277 10.838791 12.0119327  8.9257628  9.751487
> g4 10.174802 12.0263061 10.885067 13.4689705  8.0596727 11.040014
> g5 -2.175861  0.4178679 -4.190449 -1.6150923 -1.7203607 -3.463298
> g6 -1.963736 -0.8426501 -2.763640 -0.8267138  0.7331591 -1.034681
>          [,31]      [,32]      [,33]     [,34]      [,35]      [,36]
> g1 13.2845847 13.2969523 11.9704449 13.369279 12.7079258 13.4308818
> g2 13.4648886 12.6173940 11.4121843 13.404501 12.3699831  9.6093653
```

```
> g3  9.8923785 10.9311813  6.9765048 10.064224 10.0328385  8.9825370
> g4 12.1910768 11.0352071  9.4160628 10.503397  9.1322501 12.1163873
> g5  0.6864578 -3.3087704 -0.9919851 -3.938833 -4.3531769 -3.5996567
> g6 -0.7689391  0.3169013 -2.9008971 -1.622064  0.4240979 -0.7543516
>         [,37]     [,38]      [,39]      [,40]     [,41]     [,42]     [,43]
> g1 12.507077 11.780734 11.3764345 10.0284497 10.936023 13.438335 13.432059
> g2 14.333239 12.664291  9.9126366 12.4797136 11.003269 10.968907  9.756589
> g3  8.949023  9.968663  8.7287935  9.8793372  8.845232 10.831307 10.777845
> g4 12.117538 10.549089  9.6959240  8.8876984 11.457789 10.752184 10.849595
> g5 -0.653499 -1.141197 -2.7429300 -1.9058742 -1.429589 -2.109230 -1.987744
> g6 -1.669149 -1.357247 -0.5948825 -0.4377632 -3.844224  0.307805 -1.156562
>         [,44]     [,45]      [,46]      [,47]      [,48]      [,49]
> g1 14.220396 11.045483 10.4525633 12.4112332 11.8731124 13.3532400
> g2 13.157719 12.828344 11.4686264 13.8241836 10.1705312 13.5762652
> g3  8.985435  9.848462  9.9677387 12.7576262 11.8557020 10.8245602
> g4 10.342474 12.260743  8.5028642 12.0841483 11.5928861 10.9836563
> g5 -1.487860 -1.264407 -1.8873743 -2.2295309 -1.8970537 -2.8578162
> g6 -2.222222  0.264414 -0.5428139 -0.1591217 -0.1606294  0.6430459
>         [,50]      [,51]      [,52]     [,53]      [,54]     [,55]
> g1 12.878322 14.3846207 12.3644766 14.147927 12.3457363 12.470742
> g2 12.178972 13.1115180 14.4696233 12.378412 12.9770893 10.896948
> g3 10.744376  9.7798237  9.0229090  7.825118  8.7442652 10.888025
> g4 12.749827  9.7352199  8.4573819 13.252391 10.8156762 12.273182
> g5 -4.005878 -3.1768597  0.5840697 -1.254032 -1.5058959 -2.952348
> g6 -1.943017  0.5986482 -1.0459673 -3.613648  0.6278244 -2.331696
>          [,56]      [,57]        [,58]      [,59]      [,60]     [,61]
> g1 12.4782356 12.1172123  8.966316505 15.5108387 13.6376266 11.712954
> g2 12.8799712 11.1404864 10.191240048 12.4722300 10.1997121 11.378397
> g3  9.8390599  7.7544058 11.061459459  9.9583801  8.7318671 11.928056
> g4  9.3944891 10.9329711  7.857285089 13.2266156 12.6685704 10.923103
> g5 -1.3457222 -0.3536978 -0.001519137 -0.8623282 -1.2889454 -3.129713
> g6  0.0114283 -1.0998112  0.948377955 -1.3272901 -0.7624468 -1.358733
>           [,62]      [,63]     [,64]      [,65]      [,66]      [,67]
> g1 13.22425865 11.2993027 12.659673 12.4085656 12.3136012 14.1980522
> g2 10.73681985 11.7450767 12.231193  9.9584254 12.8094149 11.4629312
> g3  9.86364146  8.9322345 10.061779 10.7777676  9.9694700  9.0368804
> g4 10.32836890 10.3686335 13.198781 10.9126091 12.0236106 12.5292635
> g5 -1.95088850 -1.8012877 -1.003361  0.1851967 -0.9115779  0.1860512
> g6 -0.07970153 -0.8638046 -1.179454 -0.5993835  0.3251259 -0.6014564
>          [,68]     [,69]       [,70]     [,71]     [,72]      [,73]
> g1 13.3793006 12.125547 13.78730556  9.279926 11.744737 13.2539050
> g2 13.3949805 10.215283 14.05463402  9.892414 13.237088  8.6832292
> g3 10.8611830 10.011459  8.05483582 10.923292  9.749937  8.9834724
> g4 11.4936919  9.969241  9.86681906 11.124393  9.245135 10.0255459
> g5 -1.8415196 -1.515651 -2.65865442 -3.085898 -0.706746 -1.9337640
> g6  0.2277101  0.177062  0.04502823 -3.122996 -1.600062 -0.3827523
>         [,74]     [,75]     [,76]     [,77]      [,78]      [,79]
> g1 11.572583 12.730862 14.340484 12.067477 11.8572852 13.4582834
> g2 12.712514 13.084804 11.524907 11.858854 13.2503385 11.1871683
> g3  9.794878  8.994664 12.908469 11.918010  9.8035252  9.0500562
> g4  8.513299 11.456633 12.051300 10.866142  9.1601472 10.8102503
> g5 -1.630638  1.569939 -2.869825 -1.935004 -0.7111727 -1.8325344
> g6 -1.312369 -0.425844 -2.616749  1.517214  1.0509228 -0.9054137
```

```
>          [,80]      [,81]     [,82]     [,83]     [,84]     [,85]
> g1 12.8498839 12.5063274 15.054707 12.129779 13.782206 13.324058
> g2 12.7685025 12.8957987 11.193109  9.623507 11.097235 12.917587
> g3  9.9681949 10.0209050  8.968449 10.013932  8.848713  8.938134
> g4  9.5474666 11.8346676 11.581142  8.843743 11.398186  9.310079
> g5 -1.6724099 -0.6148641 -2.122823 -2.055773 -1.837774 -1.744670
> g6  0.1874409 -0.5571816 -1.275753  1.251321 -1.461765 -1.143176
>          [,86]      [,87]      [,88]      [,89]     [,90]      [,91]
> g1 10.8390730 12.9895794 12.3372672 13.0832621 11.442714 10.3061903
> g2 11.2496059 13.4956601 11.5297350 11.6202721 12.131359 13.5203851
> g3 10.8617391  9.9955074  7.9554000  8.0269293 10.882742  8.8139988
> g4  9.8280857 11.0754789 10.4156703 10.2620039  9.568028  8.4451421
> g5 -0.4589174  0.3783464 -0.3750307 -1.1034548 -2.414278 -0.5185182
> g6 -1.9589648 -2.6470956  0.7584896  0.8978139 -1.176699 -0.5702781
>         [,92]     [,93]     [,94]     [,95]      [,96]      [,97]
> g1 11.266927 13.494598 14.277821 13.073171 11.8758586 13.5913022
> g2 12.771425 14.685187 12.491932 11.346665 11.4054797 10.4663578
> g3  8.908053  9.015920  8.860876  9.915142  8.7825983 11.0112119
> g4 11.995117 11.916406 13.801222  9.638339 10.1347857 12.0839281
> g5 -1.405765  1.190811 -1.278458 -3.343678 -3.0078396  0.5704218
> g6 -1.525401  0.341591 -1.418298  1.282014 -0.2044016 -1.8317988
>          [,98]      [,99]     [,100]
> g1 14.2708835 13.5897961 15.5494655
> g2 12.4372747 13.4138837 10.3536817
> g3  9.9621906  9.9454910 10.9312343
> g4 12.6380722 11.2432951 12.2587410
> g5 -0.3880157 -1.1474141 -0.9004263
> g6 -3.0175926  0.2285946  0.2263821
```

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    0    1    1    1    1    2    1    1    2     1     2     2     2
> rs2    2    1    0    2    0    2    0    1    0     2     2     2     2
> rs3    0    1    2    0    1    0    2    2    1     1     0     0     2
> rs4    0    1    1    1    2    2    1    0    1     0     0     1     1
> rs5    2    2    0    2    2    1    2    2    1     1     0     1     1
> rs6    2    1    1    1    1    1    1    1    1     2     1     0     1
>     [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24]
> rs1     1     2     2     0     1     1     0     1     2     2     0
> rs2     1     1     0     1     0     1     0     2     1     0     2
> rs3     1     2     1     0     0     1     2     1     2     0     1
> rs4     1     2     1     1     1     2     1     1     0     1     1
> rs5     1     0     1     0     0     1     1     1     2     0     1
> rs6     1     2     1     1     2     0     2     1     1     1     1
>     [,25] [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35]
> rs1     1     0     1     1     0     1     0     2     1     1     2
> rs2     0     0     2     1     1     1     0     1     1     2     1
> rs3     2     1     1     2     1     0     1     2     1     0     2
> rs4     1     1     1     1     1     2     1     1     0     1     0
> rs5     1     0     0     1     1     1     1     2     1     0     2
> rs6     1     1     1     0     2     0     1     2     1     0     1
>     [,36] [,37] [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46]
```

```
> rs1     1     1     2     2     2     0     1     1     1     1     1
> rs2     1     1     1     0     1     1     1     0     2     2     1
> rs3     0     1     2     2     2     0     1     1     0     0     2
> rs4     1     0     2     0     1     1     1     1     0     1     1
> rs5     0     2     0     2     1     1     1     0     2     1     2
> rs6     2     1     2     1     1     0     2     1     1     1     1
>     [,47] [,48] [,49] [,50] [,51] [,52] [,53] [,54] [,55] [,56] [,57]
> rs1     1     1     1     1     2     2     2     2     1     1     2
> rs2     2     1     2     1     0     2     2     2     0     1     0
> rs3     2     0     0     0     0     0     1     2     2     0     1
> rs4     1     1     1     0     2     0     2     2     1     2     2
> rs5     1     1     1     1     1     0     0     1     2     1     0
> rs6     0     1     2     2     2     0     1     1     1     1     0
>     [,58] [,59] [,60] [,61] [,62] [,63] [,64] [,65] [,66] [,67] [,68]
> rs1     1     2     2     2     1     1     1     0     1     1     1
> rs2     1     0     2     1     1     0     0     0     0     2     2
> rs3     2     1     1     1     0     1     2     1     1     1     1
> rs4     0     2     0     1     1     1     1     1     1     2     0
> rs5     1     1     1     1     1     2     1     0     1     0     2
> rs6     1     1     1     1     0     0     1     1     1     2     1
>     [,69] [,70] [,71] [,72] [,73] [,74] [,75] [,76] [,77] [,78] [,79]
> rs1     1     0     2     0     1     2     0     0     1     2     1
> rs2     2     1     2     1     1     1     1     1     1     2     2
> rs3     1     2     1     2     1     2     2     2     0     0     1
> rs4     2     0     2     1     1     1     2     0     2     1     1
> rs5     1     1     1     2     1     2     2     0     0     1     1
> rs6     2     0     2     1     1     2     2     1     1     1     1
>     [,80] [,81] [,82] [,83] [,84] [,85] [,86] [,87] [,88] [,89] [,90]
> rs1     1     1     0     1     1     1     1     1     2     1     2
> rs2     0     2     1     1     1     0     0     2     1     1     0
> rs3     2     1     1     0     1     2     0     1     1     1     2
> rs4     1     0     0     1     1     1     0     2     1     0     0
> rs5     0     1     1     1     0     1     1     1     1     1     1
> rs6     0     2     1     1     0     0     1     2     2     1     2
>     [,91] [,92] [,93] [,94] [,95] [,96] [,97] [,98] [,99] [,100]
> rs1     0     1     2     1     2     1     1     1     1      1
> rs2     2     0     2     2     0     0     2     1     0      1
> rs3     2     1     1     1     0     1     1     1     1      1
> rs4     1     2     1     2     0     0     1     1     2      1
> rs5     1     0     1     1     2     1     1     1     1      0
> rs6     2     1     2     0     1     2     1     1     1      0
head(data$Data$X[[2]] - 1)
>     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
> rs1    1    1    0    1    1    1    1    2    1     2     2     0     1
> rs2    1    2    2    1    2    1    1    2    2     1     0     1     0
> rs3    2    1    2    1    2    2    2    1    0     1     2     2     0
> rs4    1    2    1    1    1    1    0    1    2     1     1     0     1
> rs5    1    1    1    1    1    0    0    2    0     1     0     1     1
> rs6    2    2    0    2    1    1    2    2    2     0     1     1     1
>     [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24]
> rs1     0     2     0     0     1     1     1     1     0     0     2
> rs2     1     2     0     1     2     0     1     1     2     2     1
> rs3     1     2     1     1     2     1     1     1     2     1     1
```

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

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] 12.028811 11.919729 11.792414 11.644675 11.474381 11.279549 11.058506
>  [8] 10.810038 10.533546 10.229192  9.898012  9.541965  9.163927  8.767605
> [15]  8.357401  7.938219  7.515262  7.093838  6.679184  6.276334  5.890012
> [22]  5.524560  5.183847  4.871179  4.589156  4.339539  4.123102  3.939537
> [29]  3.787457  3.664496  3.567537  3.493002  3.437166  3.396432  3.367533
> [36]  3.347651  3.334456  3.326095  3.321136  3.318506  3.317422  3.317322
> [43]  3.317816  3.318639  3.319614  3.320630  3.321620  3.322541  3.323374
> [50]  3.324116
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 = 194.256514, rho = 0.000000
```

```
> FSSEM@lambda = 194.256514, rho = 0.000000
> FSSEM@lambda = 194.256514, rho = 0.000000
> FSSEM@lambda = 194.256514, rho = 0.000000
> FSSEM@lambda = 194.256514, rho = 0.000000
> FSSEM@lambda = 194.256514, rho = 0.000000
> FSSEM@lambda = 194.256514, rho = 0.000000
> FSSEM@lambda = 194.256514, rho = 0.000000
> FSSEM@lambda = 194.256514, rho = 0.000000
> FSSEM@lambda = 194.256514, rho = 0.000000
> FSSEM@lambda = 90.165887, rho = 1.789177
> FSSEM@lambda = 90.165887, rho = 0.830462
> FSSEM@lambda = 90.165887, rho = 0.385466
> FSSEM@lambda = 90.165887, rho = 0.178918
> FSSEM@lambda = 90.165887, rho = 0.083046
> FSSEM@lambda = 90.165887, rho = 0.038547
> FSSEM@lambda = 90.165887, rho = 0.017892
> FSSEM@lambda = 90.165887, rho = 0.008305
> FSSEM@lambda = 90.165887, rho = 0.003855
> FSSEM@lambda = 90.165887, rho = 0.001789
> FSSEM@lambda = 41.851297, rho = 5.334570
> FSSEM@lambda = 41.851297, rho = 2.476088
> FSSEM@lambda = 41.851297, rho = 1.149298
> FSSEM@lambda = 41.851297, rho = 0.533457
> FSSEM@lambda = 41.851297, rho = 0.247609
> FSSEM@lambda = 41.851297, rho = 0.114930
> FSSEM@lambda = 41.851297, rho = 0.053346
> FSSEM@lambda = 41.851297, rho = 0.024761
> FSSEM@lambda = 41.851297, rho = 0.011493
> FSSEM@lambda = 41.851297, rho = 0.005335
> FSSEM@lambda = 19.425651, rho = 6.010180
> FSSEM@lambda = 19.425651, rho = 2.789679
> FSSEM@lambda = 19.425651, rho = 1.294854
> FSSEM@lambda = 19.425651, rho = 0.601018
> FSSEM@lambda = 19.425651, rho = 0.278968
> FSSEM@lambda = 19.425651, rho = 0.129485
> FSSEM@lambda = 19.425651, rho = 0.060102
> FSSEM@lambda = 19.425651, rho = 0.027897
> FSSEM@lambda = 19.425651, rho = 0.012949
> FSSEM@lambda = 19.425651, rho = 0.006010
> FSSEM@lambda = 9.016589, rho = 682.491856
> FSSEM@lambda = 9.016589, rho = 316.784658
> FSSEM@lambda = 9.016589, rho = 147.038413
> FSSEM@lambda = 9.016589, rho = 68.249186
> FSSEM@lambda = 9.016589, rho = 31.678466
> FSSEM@lambda = 9.016589, rho = 14.703841
> FSSEM@lambda = 9.016589, rho = 6.824919
> FSSEM@lambda = 9.016589, rho = 3.167847
> FSSEM@lambda = 9.016589, rho = 1.470384
> FSSEM@lambda = 9.016589, rho = 0.682492
> FSSEM@lambda = 4.185130, rho = 579.780365
> FSSEM@lambda = 4.185130, rho = 269.110207
> FSSEM@lambda = 4.185130, rho = 124.909893
> FSSEM@lambda = 4.185130, rho = 57.978036
```

```
> FSSEM@lambda = 4.185130, rho = 26.911021
> FSSEM@lambda = 4.185130, rho = 12.490989
> FSSEM@lambda = 4.185130, rho = 5.797804
> FSSEM@lambda = 4.185130, rho = 2.691102
> FSSEM@lambda = 4.185130, rho = 1.249099
> FSSEM@lambda = 4.185130, rho = 0.579780
> FSSEM@lambda = 1.942565, rho = 531.226478
> FSSEM@lambda = 1.942565, rho = 246.573489
> FSSEM@lambda = 1.942565, rho = 114.449275
> FSSEM@lambda = 1.942565, rho = 53.122648
> FSSEM@lambda = 1.942565, rho = 24.657349
> FSSEM@lambda = 1.942565, rho = 11.444928
> FSSEM@lambda = 1.942565, rho = 5.312265
> FSSEM@lambda = 1.942565, rho = 2.465735
> FSSEM@lambda = 1.942565, rho = 1.144493
> FSSEM@lambda = 1.942565, rho = 0.531226
> FSSEM@lambda = 0.901659, rho = 515.049664
> FSSEM@lambda = 0.901659, rho = 239.064877
> FSSEM@lambda = 0.901659, rho = 110.964086
> FSSEM@lambda = 0.901659, rho = 51.504966
> FSSEM@lambda = 0.901659, rho = 23.906488
> FSSEM@lambda = 0.901659, rho = 11.096409
> FSSEM@lambda = 0.901659, rho = 5.150497
> FSSEM@lambda = 0.901659, rho = 2.390649
> FSSEM@lambda = 0.901659, rho = 1.109641
> FSSEM@lambda = 0.901659, rho = 0.515050
> FSSEM@lambda = 0.418513, rho = 510.359917
> FSSEM@lambda = 0.418513, rho = 236.888089
> FSSEM@lambda = 0.418513, rho = 109.953711
> FSSEM@lambda = 0.418513, rho = 51.035992
> FSSEM@lambda = 0.418513, rho = 23.688809
> FSSEM@lambda = 0.418513, rho = 10.995371
> FSSEM@lambda = 0.418513, rho = 5.103599
> FSSEM@lambda = 0.418513, rho = 2.368881
> FSSEM@lambda = 0.418513, rho = 1.099537
> FSSEM@lambda = 0.418513, rho = 0.510360
> FSSEM@lambda = 0.194257, rho = 509.532862
> FSSEM@lambda = 0.194257, rho = 236.504204
> FSSEM@lambda = 0.194257, rho = 109.775527
> FSSEM@lambda = 0.194257, rho = 50.953286
> FSSEM@lambda = 0.194257, rho = 23.650420
> FSSEM@lambda = 0.194257, rho = 10.977553
> FSSEM@lambda = 0.194257, rho = 5.095329
> FSSEM@lambda = 0.194257, rho = 2.365042
> FSSEM@lambda = 0.194257, rho = 1.097755
> FSSEM@lambda = 0.194257, rho = 0.509533

fit <- fitOpt$fit
```

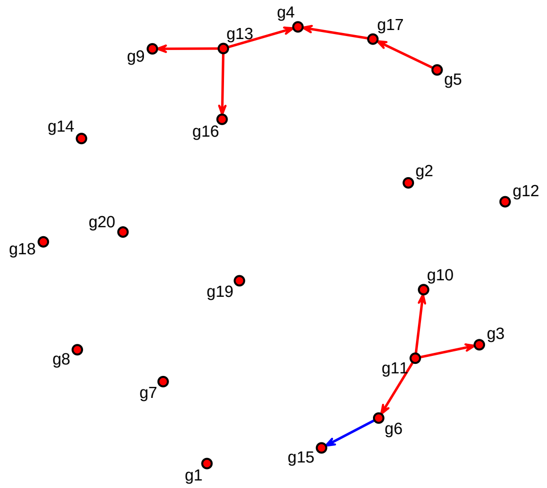

Figure 4: estimated differential GRN by fssemR

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 . . .  .         .  .         . . .
>  [2,] . . . .  .         .         0 . . 0  .         .  .         . . .
>  [3,] . . . .  .         .         . . . . -0.2187958 .  .         . . .
>  [4,] 0 . . .  .         0.0000000 . . . .  .         . -0.2096113 . . 0
>  [5,] . . . .  .         .         . . . .  .         .  .         . . .
>  [6,] . . . .  .         .         . . . . -0.3020784 .  .         0 . .
>  [7,] . . . .  .         0.0000000 . . . .  .         .  .         . . .
>  [8,] . . . .  0.0000000 .         . . . .  .         .  .         . . .
>  [9,] . . . .  .         .         . . . .  .         . -0.2580067 . . .
> [10,] . . . .  .         .         . . . . -0.2433023 .  .         . . .
> [11,] . . . .  .         .         . . . .  .         .  .         . . .
> [12,] . . . .  .         .         . . . .  .         .  .         . . .
> [13,] . . . .  0.0000000 0.0000000 . . . .  .         .  .         . . .
> [14,] . . . .  .         .         . . . .  .         .  .         . . .
> [15,] . . . 0  .         0.2281523 . . . .  .         .  .         . . .
> [16,] 0 . . .  .         .         . . . .  .         . -0.3450633 . . .
> [17,] . . . . -0.2116012 0.0000000 . . . .  .         .  .         0 . .
> [18,] . . . .  .         .         . . . .  0.0000000 .  .         . . .
> [19,] 0 . . .  .         .         . . . .  .         .  .         . . .
> [20,] . . . .  .         .         . . . .  .         .  .         . . .
> 
>  [1,]  .         . . .
>  [2,]  .         . . .
>  [3,]  .         . . .
>  [4,] -0.1747657 0 . .
>  [5,]  .         . 0 .
>  [6,]  .         . . .
>  [7,]  .         . . .
>  [8,]  .         . . .
>  [9,]  .         . 0 .
> [10,]  .         . . .
> [11,]  .         . . .
> [12,]  0.0000000 . . .
> [13,]  .         . . .
> [14,]  .         . . .
> [15,]  0.0000000 . . .
> [16,]  .         . . .
> [17,]  .         . . .
> [18,]  .         . . .
> [19,]  .         . . .
> [20,]  .         . . .
```

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=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_1.2-14      ggnetwork_0.5.1    ggplot2_3.2.0.9000
> [4] network_1.13.0.1   fssemR_0.1.6
>
> loaded via a namespace (and not attached):
>  [1] Rcpp_1.0.0           sna_2.4              bindr_0.1.1
>  [4] compiler_3.4.0       pillar_1.3.1         iterators_1.0.9
>  [7] tools_3.4.0          digest_0.6.18        evaluate_0.12
> [10] tibble_2.0.1         gtable_0.2.0         lattice_0.20-35
> [13] pkgconfig_2.0.2      rlang_0.3.1          foreach_1.4.4
> [16] igraph_1.2.2         ggrepel_0.8.0        yaml_2.2.0
> [19] parallel_3.4.0       mvtnorm_1.0-8        xfun_0.4
> [22] bindrcpp_0.2.2       coda_0.19-2          withr_2.1.2
> [25] stringr_1.3.1        dplyr_0.7.8          knitr_1.21
> [28] tidyselect_0.2.5     glmnet_2.0-16        grid_3.4.0
> [31] glue_1.3.0           R6_2.2.2             qtl_1.44-9
> [34] rmarkdown_1.11       purrr_0.3.0          magrittr_1.5
> [37] scales_1.0.0.9000    codetools_0.2-15     htmltools_0.3.6
> [40] MASS_7.3-49          assertthat_0.2.0     colorspace_1.4-0
> [43] stringi_1.2.4        lazyeval_0.2.1       munsell_0.5.0
> [46] statnet.common_4.1.4 crayon_1.3.4
```