Escherichia Coli Network

Example for GeneNet 1.2.13 (August 2015) or later

This note reproduces the "Escherichia coli" network example from J. Schäfer and K. Strimmer. 2005. *A shrinkage approach to large-scale covariance estimation and implications for functional genomics.* Statist. Appl. Genet. Mol. Biol. **4**: 32 (http://dx.doi.org/10.2202/1544-6115.1175)

Load GeneNet package

```
library("GeneNet")
```

```
## Loading required package: corpcor
## Loading required package: longitudinal
## Loading required package: fdrtool
```

E. Coli data set (9 time points for 102 genes):

```
data(ecoli)
dim(ecoli)
```

```
## [1]   9 102
```

Estimation of partial correlations

Estimate matrix of partial correlation using a shrinkage estimator:

```
pc = ggm.estimate.pcor(ecoli)
```

```
## Estimating optimal shrinkage intensity lambda (correlation matrix): 0.1804
```

```
dim(pc)
```

```
## [1] 102 102
```

Assign p-values, q-values and empirical posterior probabilities to all 5151 potential edges in the network:

```
ecoli.edges = network.test.edges(pc, direct=TRUE, fdr=TRUE)
```

```
## Estimate (local) false discovery rates (partial correlations):
## Step 1... determine cutoff point
## Step 2... estimate parameters of null distribution and eta0
## Step 3... compute p-values and estimate empirical PDF/CDF
## Step 4... compute q-values and local fdr
## Step 5... prepare for plotting
```

```
##
## Estimate (local) false discovery rates (log ratio of spvars):
## Step 1... determine cutoff point
## Step 2... estimate parameters of null distribution and eta0
## Step 3... compute p-values and estimate empirical PDF/CDF
## Step 4... compute q-values and local fdr
## Step 5... prepare for plotting
```

```
dim(ecoli.edges)
```

```
## [1] 5151   10
```

The table lists all edges in the order strength of partial correlations:

```
ecoli.edges[1:5,]
```

```
##         pcor node1 node2         pval         qval      prob    log.spvar
## 1  0.2318566    51    53 2.220446e-16 3.612205e-13 1.0000000 -0.043537019
## 2  0.2240555    52    53 2.220446e-16 3.612205e-13 1.0000000 -0.040249854
## 3  0.2150782    51    52 2.220446e-16 3.612205e-13 1.0000000 -0.003287165
## 4  0.1732886     7    93 3.108624e-15 3.792816e-12 0.9999945 -0.025293430
## 5 -0.1341889    29    86 1.120811e-09 1.093997e-06 0.9999516  0.022305368
##    pval.dir  qval.dir     prob.dir
## 1 0.3803869 0.7557272 2.220446e-16
## 2 0.4173922 0.7724561 2.220446e-16
## 3 0.9471949 0.8851073 2.220446e-16
## 4 0.6103234 0.8323249 2.220446e-16
## 5 0.6531371 0.8415749 2.220446e-16
```

Decide which edges to include in the network

To obtain a graph you need to select top ranking edges according to a suitable criterion. Here are some suggestions:

1. Use local fdr cutoff 0.2, i.e. include all edges with posterior probability of at least 0.8.

```
ecoli.net = extract.network(ecoli.edges)
```

```
##
## Significant edges:  125
##     Corresponding to  2.43 %  of possible edges
##
## Significant directions:  377
##     Corresponding to  7.32 %  of possible directions
## Significant directions in the network:  17
##     Corresponding to  13.6 %  of possible directions in the network
```

```
dim(ecoli.net)
```

```
## [1] 125  11
```

2. Use local fdr cutoff 0.1, i.e. i.e. include all edges with posterior probability of at least 0.9.

```
ecoli.net = extract.network(ecoli.edges, cutoff.ggm=0.9, cutoff.dir=0.9)
```

```
##
## Significant edges:  65
##     Corresponding to  1.26 %  of possible edges
##
## Significant directions:  269
##     Corresponding to  5.22 %  of possible directions
## Significant directions in the network:  6
##     Corresponding to  9.23 %  of possible directions in the network
```

```
dim(ecoli.net)
```

```
## [1] 65 11
```

3. Include a fixed number of edges, say the 70 strongest edges

```
ecoli.net = extract.network(ecoli.edges, method.ggm="number", cutoff.ggm=70)
```

```
##
## Significant edges:  70
##     Corresponding to  1.36 %  of possible edges
##
## Significant directions:  377
##     Corresponding to  7.32 %  of possible directions
## Significant directions in the network:  9
##     Corresponding to  12.86 %  of possible directions in the network
```

```
dim(ecoli.net)
```

```
## [1] 70 11
```

Plot network For plotting we use the graph and Rgraphviz packages from Bioconductor.

```
library("Rgraphviz")
```

```
## Loading required package: graph
## Loading required package: grid
```

Create graph object from the list of edges:

```
node.labels = colnames(ecoli)
gr = network.make.graph(ecoli.net, node.labels, drop.singles=TRUE)
table(  edge.info(gr)$dir )
```

```
##
## forward    none
##       9      61
```

```
sort( node.degree(gr), decreasing=TRUE)
```

```
##  sucA  cspG  fixC  yheI  lacA  lacY  lacZ  asnA  eutG  yceP  yedE  ygcE
##    11     8     7     7     6     6     6     5     5     5     5     5
##  pspA  atpD b1191 b1583  cspA  icdA  mopB  pspB  tnaA  yaeM  ycgX  yfaD
##     4     3     3     3     3     3     3     3     3     3     3     3
##  dnaG  dnaK  hupB  ibpB  yfiA  aceB  atpG b1963  cchB  dnaJ  flgD  folK
##     2     2     2     2     2     1     1     1     1     1     1     1
##  ftsJ  gltA  lpdA  nmpC  nuoM  sucD  yecO  ygbD  yhdM  yjbO
##     1     1     1     1     1     1     1     1     1     1
```

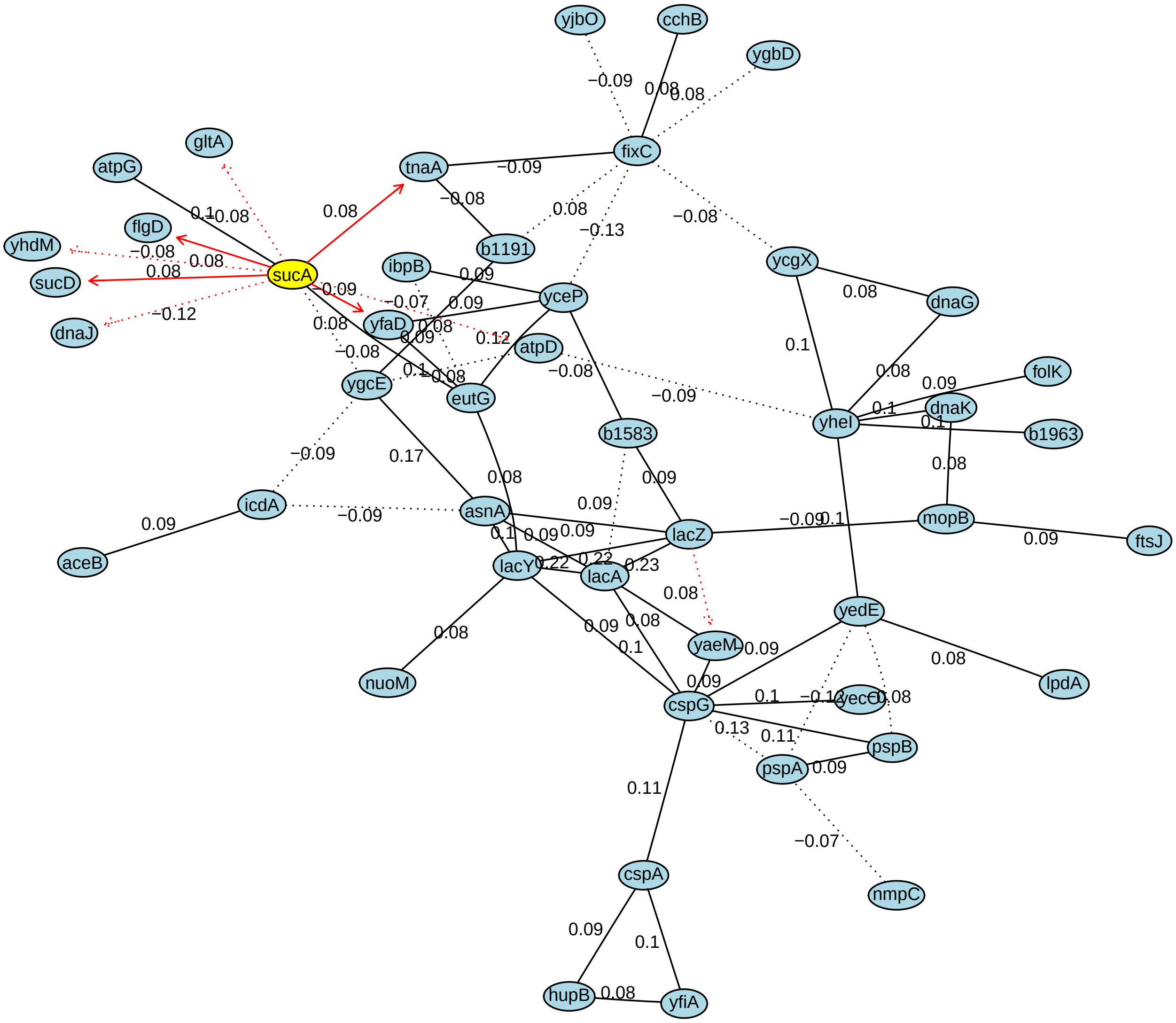
Set node and edge attributes for more beautiful graph plotting:

```
globalAttrs = list()
globalAttrs$edge = list(color = "black", lty = "solid", lwd = 1, arrowsize=1)
globalAttrs$node = list(fillcolor = "lightblue", shape = "ellipse", fixedsize = FALSE)

nodeAttrs = list()
nodeAttrs$fillcolor = c('sucA' = "yellow")

edi = edge.info(gr)
edgeAttrs = list()
edgeAttrs$dir = edi$dir # set edge directions
edgeAttrs$lty = ifelse(edi$weight < 0, "dotted", "solid") # negative correlation -> dotted
edgeAttrs$color = ifelse(edi$dir == "none", "black", "red")
edgeAttrs$label = round(edi$weight, 2) # use partial correlation as edge labels
```

```
plot(gr, attrs = globalAttrs, nodeAttrs = nodeAttrs, edgeAttrs = edgeAttrs, "fdp")
```


yjbO
cchB
ygbD
−0.09
0.08
0.08
gltA
fixC
atpG
tnaA
−0.09
0.08
−0.08
0.1
−0.08
0.08
−0.08
flgD
−0.13
yhdM
−0.08
b1191
0.08
ycgX
sucA
ibpB
0.09
sucD
0.08
−0.09
yceP
dnaG
−0.07
0.09
0.08
0.08
yfaD
0.08
dnaJ
−0.12
0.09
0.12
atpD
0.1
−0.08
0.08
folK
−0.08
−0.08
0.09
ygcE
0.1
eutG
dnaK
−0.09
0.1
yheI
b1963
b1583
−0.09
0.17
0.08
0.09
0.08
icdA
0.09
0.09
asnA
−0.09
−0.09
0.1
mopB
0.09
aceB
0.1
0.09
lacZ
ftsJ
0.09
lacY
0.22
0.22
0.23
lacA
0.08
yedE
0.08
0.08
0.09
0.08
yaeM
0.09
lpdA
0.1
nuoM
0.09
cspG
0.1
−0.12
yecO
−0.08
0.13
0.11
pspB
pspA
0.09
0.11
−0.07
cspA
nmpC
0.09
0.1
hupB
0.08
yfiA