SANTAnotebook.R

maneat

Fri Oct 30 12:33:12 2015

```
# generate the simulated network
require(SANTA)
```

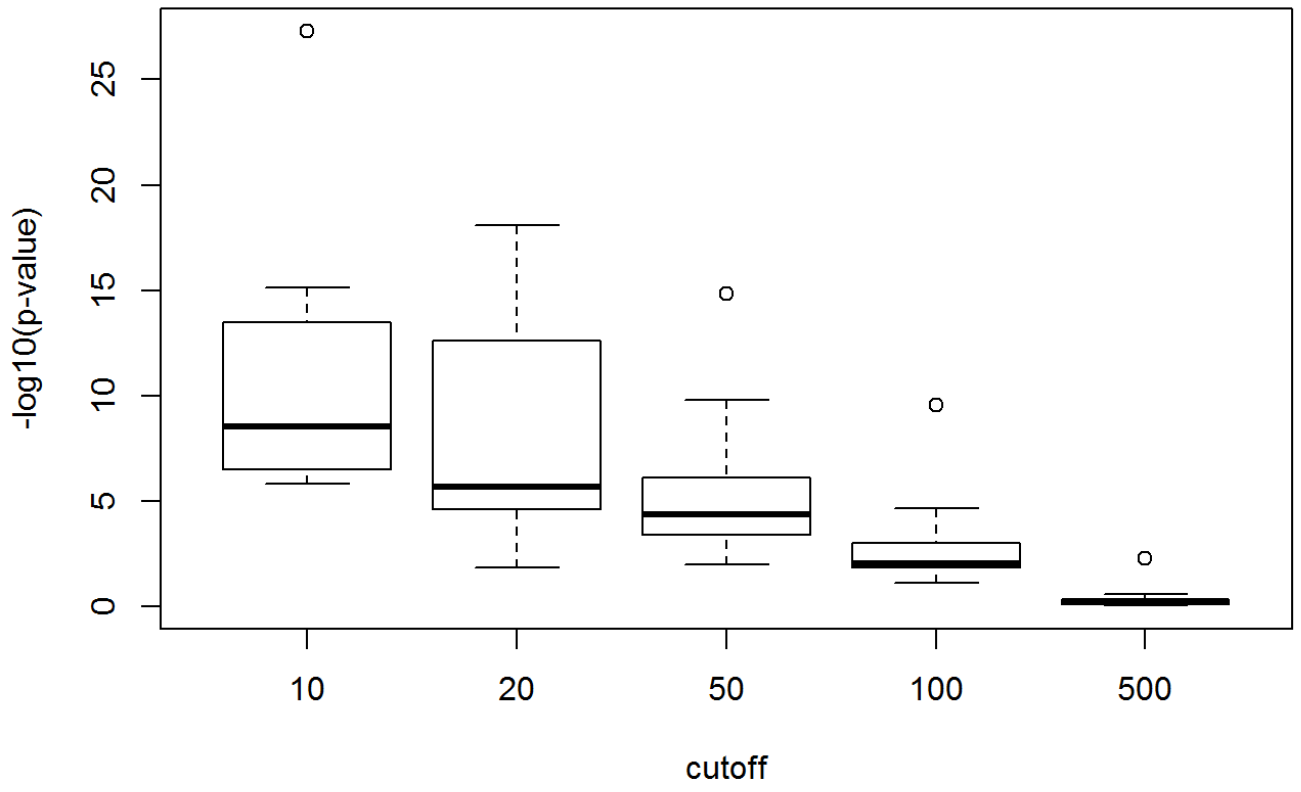
```
## Loading required package: SANTA
## Loading required package: igraph
##
## Attaching package: 'igraph'
##
## The following objects are masked from 'package:stats':
##
##     decompose, spectrum
##
## The following object is masked from 'package:base':
##
##     union
```

```
require(igraph)
set.seed(1) # for reproducibility
g <- barabasi.game(n=500, power=1, m=1, directed=F)
```

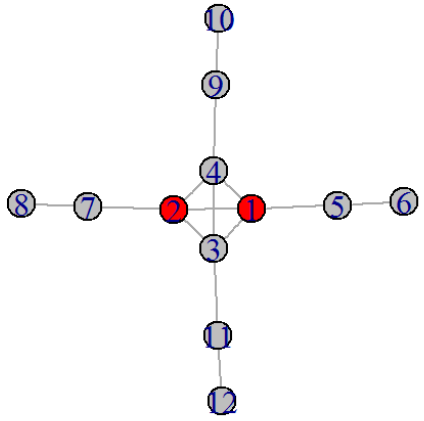
```
# measure the distance between pairs of vertices in g
dist.method <- "shortest.paths"
D <- DistGraph(g, dist.method=dist.method, verbose=F)
```

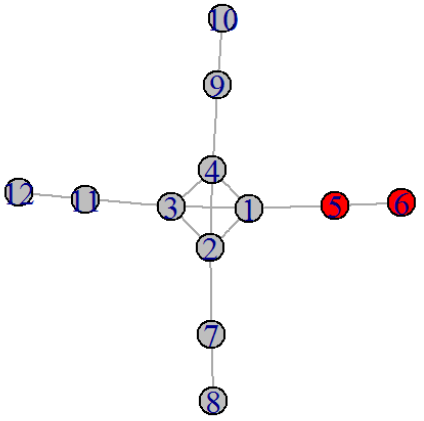
```
# place the distances into discreet bins
B <- BinGraph(D, verbose=F)
```

```
cluster.size <- 5
s.to.use <- c(10, 20, 50, 100, 500)
n.trials <- 10
pvalues <- array(0, dim=c(n.trials, length(s.to.use)),
                 dimnames=list(NULL, as.character(s.to.use)))

# run the trials for each value of s
for (s in s.to.use) {
  for (i in 1:n.trials) {
    # generate the hit set
    seed.vertex <- sample(vcount(g), 1) # identify seed
    sample.set <- order(D[seed.vertex, ])[1:s]
    hit.set <- sample(sample.set, cluster.size)

    # measure the stength of association
    g <- set.vertex.attribute(g, name="hits",
                              value=as.numeric(1:vcount(g) %in% hit.set))
    pvalues[i, as.character(s)] <- Knet(g, nperm=100,
                                        dist.method=dist.method, vertex.attr="hits",
                                        B=B, verbose=F)$pval
  }
}
```



```
# create the network
n.nodes <- 12
edges <- c(1,2, 1,3, 1,4, 2,3, 2,4, 3,4, 1,5, 5,6,
           2,7, 7,8, 4,9, 9,10, 3,11, 11,12)
weights1 <- weights2 <- rep(0, n.nodes)
weights1[c(1,2)] <- 1
weights2[c(5,6)] <- 1

g <- graph.empty(n.nodes, directed=F)
g <- add.edges(g, edges)
g <- set.vertex.attribute(g, "weights1", value=weights1)
g <- set.vertex.attribute(g, "weights2", value=weights2)
```






```
# set 1
Knet(g, nperm=100, vertex.attr="weights1", verbose=F)$pval
```

```
## [1] 0.3986885
```

```
Compactness(g, nperm=100, vertex.attr="weights1", verbose=F)$pval
```

```
## [1] 0.08767958
```

```
# set 2
Knet(g, nperm=100, vertex.attr="weights2", verbose=F)$pval
```

```
## [1] 0.002557689
```

```
Compactness(g, nperm=100, vertex.attr="weights2", verbose=F)$pval
```

```
## [1] 0.09822864
```

```
# load igraph objects
data(g.costanzo.raw)
data(g.costanzo.cor)
networks <- list(raw=g.costanzo.raw, cor=g.costanzo.cor)
network.names <- names(networks)
network.genes <- V(networks$raw)$name
```

```
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
```

```
# genes identical across networks
```

```
# obtain the GO term associations from org.Sc.sgd.db package
library(org.Sc.sgd.db)
```

```
## Loading required package: AnnotationDbi
## Loading required package: stats4
## Loading required package: BiocGenerics
## Loading required package: parallel
##
## Attaching package: 'BiocGenerics'
##
## The following objects are masked from 'package:parallel':
##
##     clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##     clusterExport, clusterMap, parApply, parCapply, parLapply,
##     parLapplyLB, parRapply, parSapply, parSapplyLB
##
## The following objects are masked from 'package:igraph':
##
##     normalize, union
##
## The following objects are masked from 'package:stats':
##
##     IQR, mad, xtabs
##
## The following objects are masked from 'package:base':
##
##     anyDuplicated, append, as.data.frame, as.vector, cbind,
##     colnames, do.call, duplicated, eval, evalq, Filter, Find, get,
##     grep, grepl, intersect, is.unsorted, lapply, lengths, Map,
##     mapply, match, mget, order, paste, pmax, pmax.int, pmin,
##     pmin.int, Position, rank, rbind, Reduce, rownames, sapply,
##     setdiff, sort, table, tapply, union, unique, unlist, unsplit
##
## Loading required package: Biobase
## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.
##
## Loading required package: IRanges
## Loading required package: S4Vectors
##
## Attaching package: 'S4Vectors'
##
## The following object is masked from 'package:igraph':
##
##     compare
##
##
## Attaching package: 'IRanges'
##
## The following object is masked from 'package:igraph':
##
##     simplify
##
```

```
## Loading required package: DBI
```

```
xx <- as.list(org.Sc.sgdGO2ALLORFS)
go.terms <- c("GO:0000082", "GO:0003682", "GO:0007265",
              "GO:0040008", "GO:0090329")
# apply the GO terms to the networks
for (name in network.names) {
  for (go.term in go.terms) {
    networks[[name]] <- set.vertex.attribute(
      networks[[name]], name=go.term,
      value=as.numeric(network.genes %in% xx[[go.term]]))
  }
}
```

```
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
```

```
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
```

```
results <- list()
for (name in network.names) {
     results[[name]] <- Knet(networks[[name]], nperm=1000,
         vertex.attr=go.terms, edge.attr="distance", verbose=F)
     results[[name]] <- sapply(results[[name]],
         function(res) res$pval)
 }
```

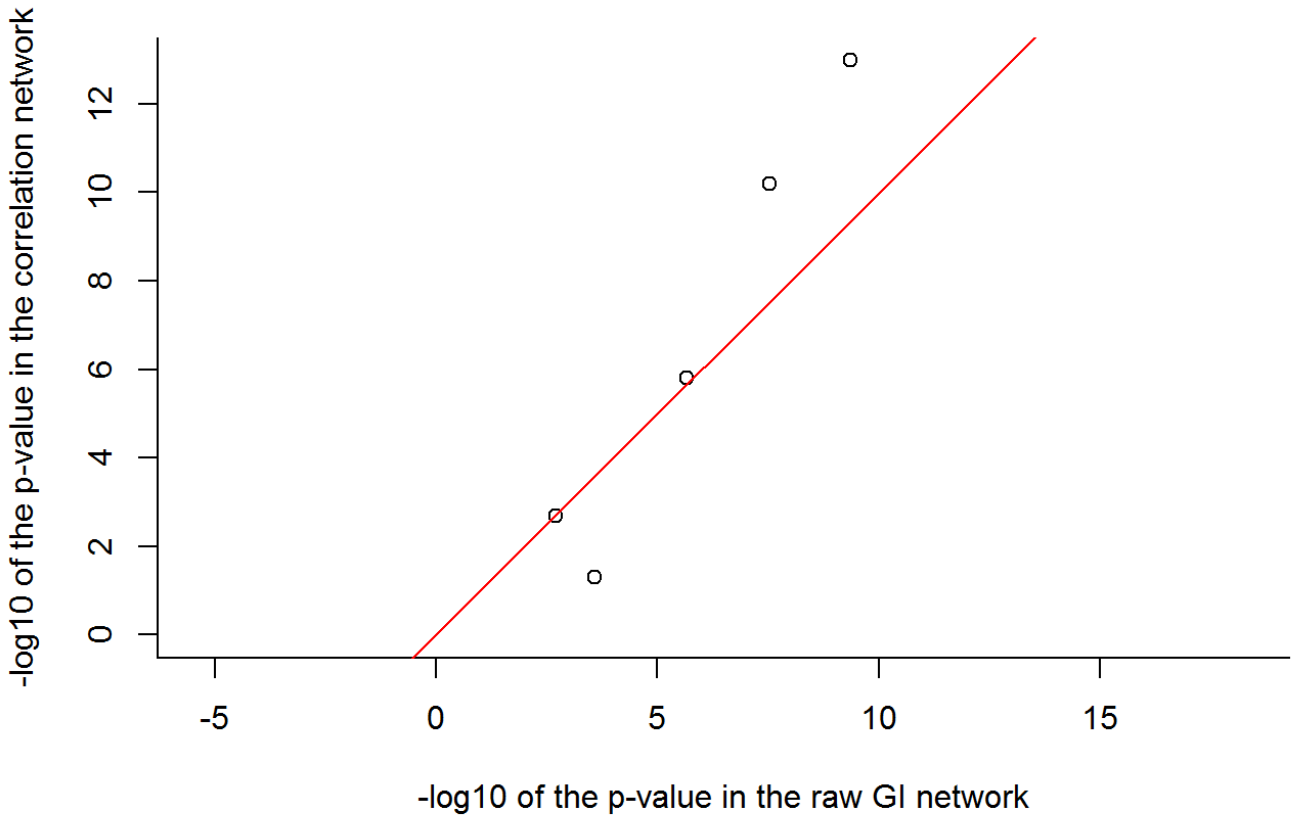
```
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
```

```
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
```

```
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
```

```
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
```

```
p.values <- array(unlist(results), dim=c(length(go.terms),
      length(network.names)), dimnames=list(go.terms,
      network.names))
p.values.ml10 <- -log10(p.values)
axis.range <- c(0, max(p.values.ml10))
plot(p.values.ml10[, "raw"], p.values.ml10[, "cor"], asp=1,
      xlim=axis.range, ylim=axis.range, bty="l",
      xlab="-log10 of the p-value in the raw GI network",
      ylab="-log10 of the p-value in the correlation network",
      main="")
abline(0, 1, col="red")
```




```
# load igraph objects
data(g.bandyopadhyay.treated)
data(g.bandyopadhyay.untreated)
networks <- list(
  treated=g.bandyopadhyay.treated,
  untreated=g.bandyopadhyay.untreated
)
network.names <- names(networks)
```

```
# obtain GO term associations
library(org.Sc.sgd.db)
xx <- as.list(org.Sc.sgdGO2ALLORFS)
# change to use alternative GO terms
associated.genes <- xx[["GO:0006974"]]
associations <- sapply(networks, function(g)
  as.numeric(V(g)$name %in% associated.genes),
  simplify=F)
networks <- sapply(network.names, function(name)
  set.vertex.attribute(networks[[name]], "rdds",
                       value=associations[[name]]), simplify=F)
```

```
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
```

```
results <- sapply(networks, function(g) Knet(g, nperm=1000,
    dist.method="shortest.paths", vertex.attr="rdds",
    edge.attr="distance"), simplify=F)
```

```
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## computing graph distance matrix... This graph was created by an old(er) igraph versi
on.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
```

```
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## done
## computing graph distance bins... done
## computing the clustering of the 'rdds' weights using 1000 permutations...  done
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
```

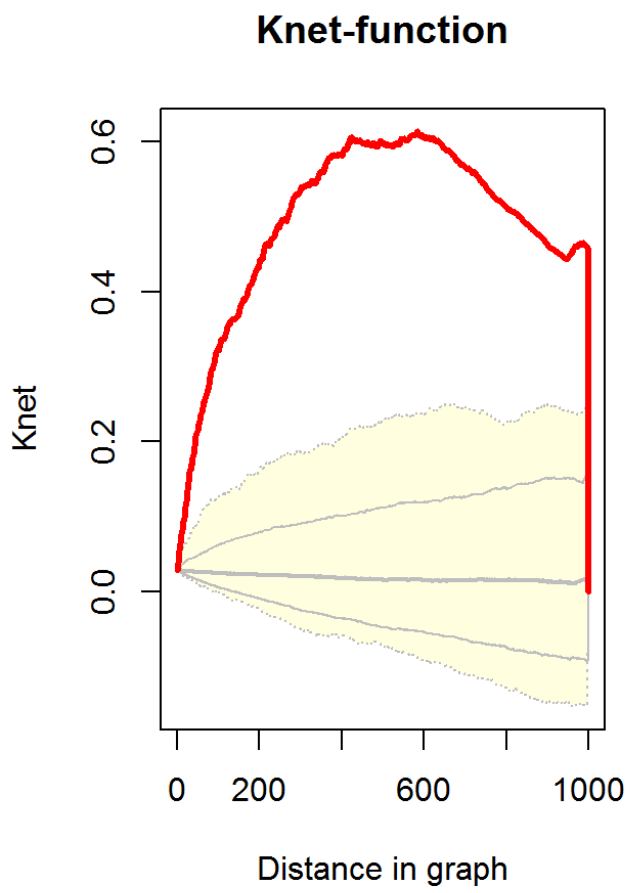
```
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## computing graph distance matrix... This graph was created by an old(er) igraph versi
on.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
```

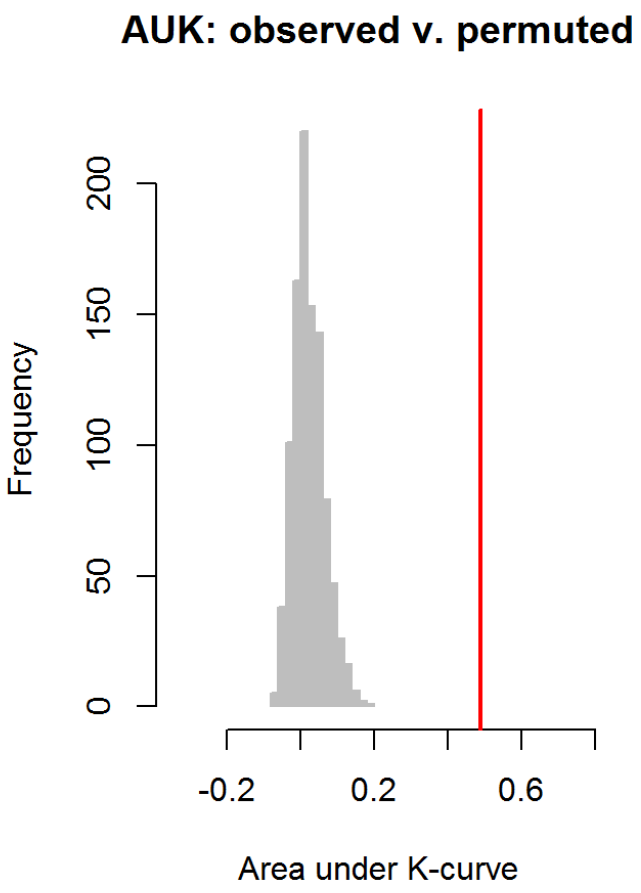
```
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## done
## computing graph distance bins... done
## computing the clustering of the 'rdds' weights using 1000 permutations...  done
```

```
p.values <- sapply(results, function(res) res$pval)
p.values
```

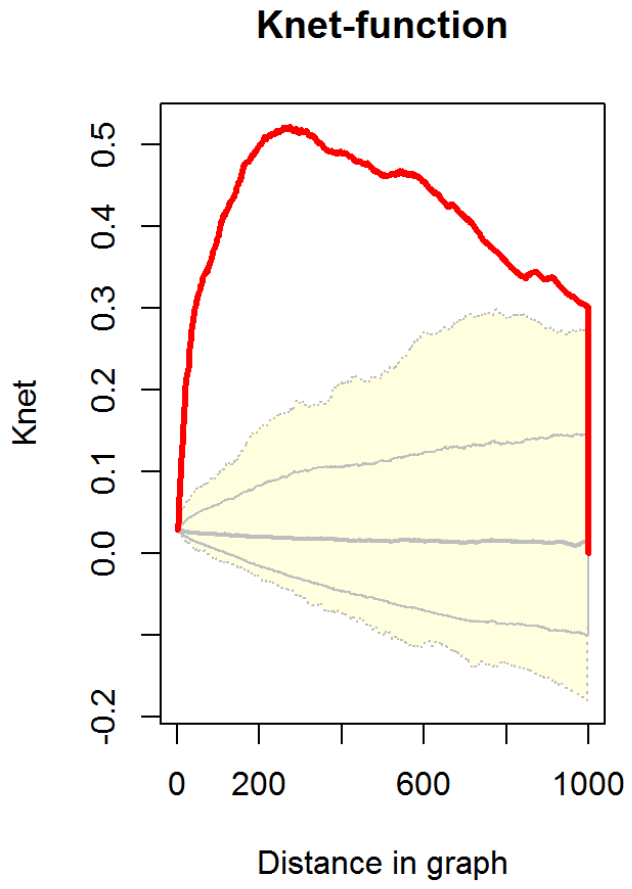
```
##      treated    untreated
## 5.857237e-29 9.357208e-19
```

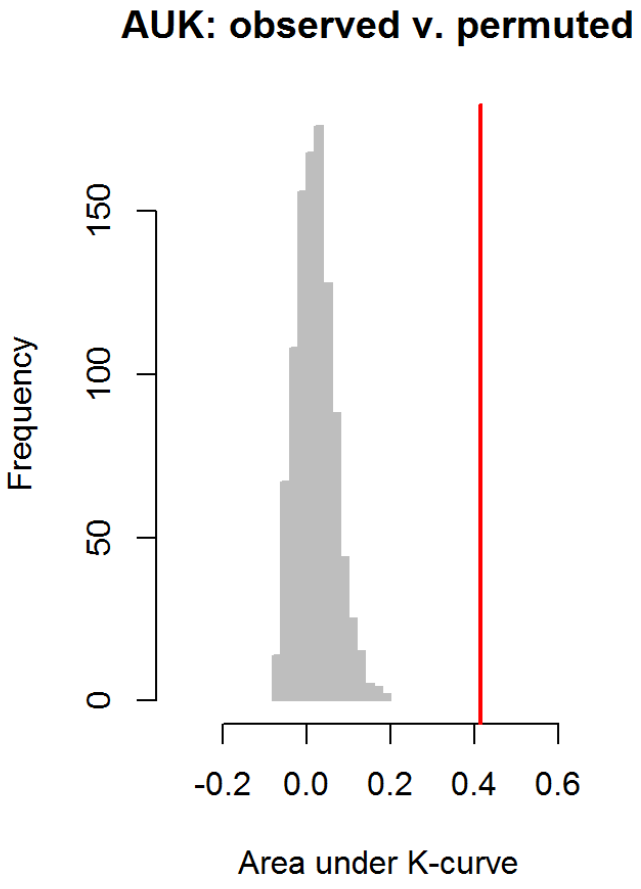
```
plot(results$treated)
```






```
plot(results$untreated)
```






```
# laod igraph object
data(g.srivas.high)
data(g.srivas.untreated)
networks <- list(
  high=g.srivas.high,
  untreated=g.srivas.untreated
)
network.names <- names(networks)
```

```
# obtain GO term associations
library(org.Sc.sgd.db)
xx <- as.list(org.Sc.sgdGO2ALLORFS)
associated.genes <- xx[["GO:0000725"]]
associations <- sapply(networks, function(g)
  as.numeric(V(g)$name %in% associated.genes),
  simplify=F)
networks <- sapply(network.names, function(name)
  set.vertex.attribute(networks[[name]], "dsbr",
                       value=associations[[name]]), simplify=F)
```

```
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
```

```
p.values <- sapply(networks, function(g)
      Knet(g, nperm=1000, dist.method="shortest.paths",
      vertex.attr="dsbr", edge.attr="distance")$pval)
```

```
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## computing graph distance matrix... This graph was created by an old(er) igraph versi
on.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
```

```
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## done
## computing graph distance bins... done
## computing the clustering of the 'dsbr' weights using 1000 permutations...  done
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
```

```
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## computing graph distance matrix... This graph was created by an old(er) igraph versi
on.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
```

```
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## This graph was created by an old(er) igraph version.
##   Call upgrade_graph() on it to use with the current igraph version
##   For now we convert it on the fly...
## done
## computing graph distance bins... done
## computing the clustering of the 'dsbr' weights using 1000 permutations...  done
```

```
p.values
```

```
##         high    untreated
## 3.197017e-06 5.568747e-01
```

```
# import and convert RNAi data
data(rnai.cheung)
rnai.cheung <- -log10(rnai.cheung)
rnai.cheung[!is.finite(rnai.cheung)] <- max(rnai.cheung[is.finite(rnai.cheung)])
rnai.cheung[rnai.cheung < 0] <- 0
```

```
# import and create IntAct network
data(edgelist.intact)
g.intact <- graph.edgelist(as.matrix(edgelist.intact),
                           directed=FALSE)

# import data and create HumanNet network
data(edgelist.humannet)
g.humannet <- graph.edgelist(as.matrix(edgelist.humannet)[,1:2],
                             directed=FALSE)
g.humannet <- set.edge.attribute(g.humannet, "distance",
                                 value=edgelist.humannet$distance)
networks <- list(intact=g.intact, humannet=g.humannet)
```

```
network.names <- names(networks)
network.genes <- sapply(networks, get.vertex.attribute,
                        name="name", simplify=F)
rnai.cheung.genes <- rownames(rnai.cheung)
cancers <- colnames(rnai.cheung)

for (cancer in cancers) {
  for (name in network.names) {
    vertex.weights <-rep(NA, vcount(networks[[name]]))
    names(vertex.weights) <- network.genes[[name]]
    common.genes <- rnai.cheung.genes[rnai.cheung.genes
                                      %in% network.genes[[name]]]
    vertex.weights[common.genes] <- rnai.cheung[common.genes, cancer]
    networks[[name]] <- set.vertex.attribute(networks[[name]],
                                             cancer, value=vertex.weights)
  }
}
```

```
#knet.res <- sapply(networks, Knet, nperm=100,
 #  dist.method="shortest.paths", vertex.attr=cancers,
  # edge.attr="distance", simplify=F)
#p.values <- sapply(knet.res, function(i) sapply(i,
 #  function(j) j$pval))
```

```
library(BioNet)
```

```
## Loading required package: graph
##
## Attaching package: 'graph'
##
## The following objects are masked from 'package:igraph':
##
##     degree, edges, intersection
##
## Loading required package: RBGL
##
## Attaching package: 'RBGL'
##
## The following objects are masked from 'package:igraph':
##
##     bfs, dfs, transitivity
```

```
# # required parameters
 n.nodes <- 1000
 n.hits <- 10
 clusters <- 3
#
# # create network and spread hits across 3 clusters
 g <- barabasi.game(n=n.nodes, power=1, m=1, directed=FALSE)
 g <- SpreadHits(g, h=n.hits, clusters=clusters, distance.cutoff=12,
     lambda=10, dist.method="shortest.paths", verbose=FALSE)
```

```
# # simulate p-values
 library(msm)
 hits <- which(get.vertex.attribute(g, "hits") == 1)
 p.values <- runif(vcount(g))
 names(p.values) <- as.character(1:vcount(g))
 p.values[as.character(hits)] <- rtnorm(n.hits * clusters, mean=0,
     sd=10e-6, lower=0, upper=1)
```

```
# # apply BioNet to the network and p-values
 bum <- fitBumModel(p.values, plot=F)
 scores <- scoreNodes(network=g, fb=bum, fdr=0.1)
 module <- runFastHeinz(g, scores)
```

```
# # apply Knode to the network
 g <- set.vertex.attribute(g, name="pheno", value=-log10(p.values))
 knode.results <- Knode(g, dist.method="diffusion",
     vertex.attr="pheno", verbose=FALSE)
```

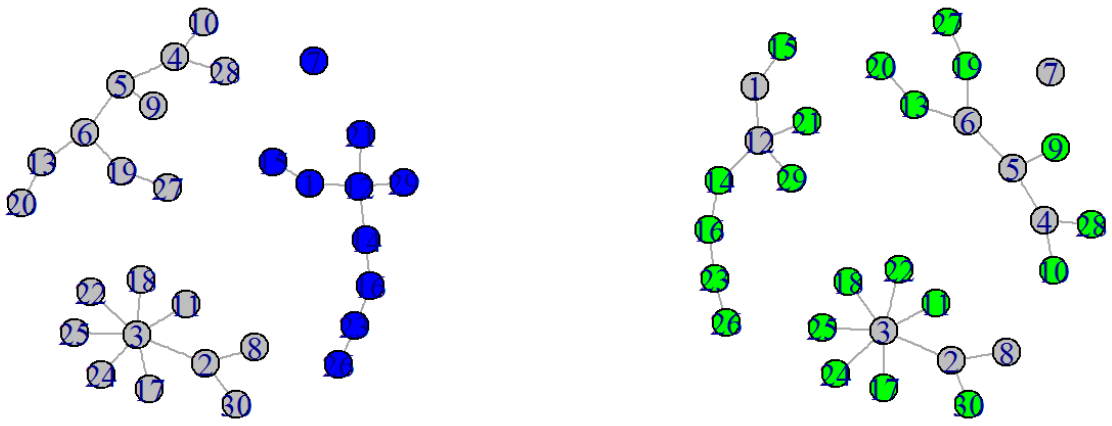
```
# # number of hits identified by BioNet
 sum(hits %in% as.numeric(V(module)$name))
```

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

```
#
# # number of hits ranked within the top 30 by Knode
 sum(hits %in% as.numeric(names(knode.results)[1:(n.hits * clusters)]))
```

```
## [1] 21
```

```
# # create subnetworks
 g.bionet <- g.knode <- induced.subgraph(g, hits)
 color.bionet <- color.knode <- rep("grey", vcount(g.bionet))
 color.bionet[hits %in% as.numeric(V(module)$name)] <- "blue"
 color.knode[hits %in% as.numeric(names(knode.results)[1:(n.hits * clusters)])] <- "gre
en"
 g.bionet <- set.vertex.attribute(g.bionet, "color", value=color.bionet)
 g.knode <- set.vertex.attribute(g.knode, "color", value=color.knode)
#
# # plot subnetworks
 par(mfrow=c(1,2))
 plot(g.bionet)
 plot(g.knode)
```



```
# # load required package
 library(SANTA)
 library(BioNet)
 library(DLBCL)
 data(exprLym)
 data(dataLym)
 data(interactome)
```

```
# # extract entrez IDs
 library(stringr)
```

```
##
## Attaching package: 'stringr'
##
## The following object is masked from 'package:graph':
##
##     boundary
##
## The following object is masked from 'package:igraph':
##
##     %>%
```

```
#
# # aggregate p-values
 pvals <- cbind(dataLym$t.pval, dataLym$s.pval)
 pval <- aggrPvals(pvals, order=2, plot=F)
 names(pval) <- dataLym$label
```

```
# # derive Lymphochip-specific network
 #network <- subNetwork(featureNames(exprLym), interactome)
 #network <- largestComp(network) # use only the largest component
 #network <- igraph.from.graphNEL(network, name=T, weight=T)
 #network <- simplify(network)
```

```
# # run BioNet on the Lymphochip-specific network and aggregate p-values
# fb <- fitBumModel(pval, plot=F)
 #scores <- scoreNodes(network, fb, fdr=0.001)
 #module <- runFastHeinz(network, scores)
 #extract.entrez <- function(x) str_extract(str_extract(x,
 #  "[(][0-9]+"), "[0-9]+")
# bionet.genes <- extract.entrez(V(module)$name)
```

```
# # convert p-values to vertex weights
 #vertex.weights <- -log10(pval)[get.vertex.attribute(network,
   #"name")]
 #network <- set.vertex.attribute(network, name="pheno",
   #value=vertex.weights)
#
# # run Knode on the Lymphochip-specific network and
# # converted aggregate p-values
 #knode.results <- Knode(network, dist.method="diffusion",
  #   vertex.attr="pheno", verbose=F)
 #knode.genes <- extract.entrez(names(knode.results)[1:vcount(module)])
```

```
 #data(go.entrez)
 #sum(go.entrez %in% bionet.genes)
 #sum(go.entrez %in% knode.genes)
```

```
sessionInfo()
```

```
## R version 3.2.2 (2015-08-14)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 7 x64 (build 7601) Service Pack 1
##
## locale:
## [1] LC_COLLATE=English_United Kingdom.1252
## [2] LC_CTYPE=English_United Kingdom.1252
## [3] LC_MONETARY=English_United Kingdom.1252
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United Kingdom.1252
##
## attached base packages:
## [1] parallel  stats4    stats     graphics  grDevices utils     datasets
## [8] methods   base
##
## other attached packages:
##  [1] stringr_1.0.0        DLBCL_1.10.0         msm_1.5
##  [4] BioNet_1.30.0        RBGL_1.46.0          graph_1.48.0
##  [7] org.Sc.sgd.db_3.2.3  RSQLite_1.0.0        DBI_0.3.1
## [10] AnnotationDbi_1.32.0 IRanges_2.4.1        S4Vectors_0.8.0
## [13] Biobase_2.30.0       BiocGenerics_0.16.0  SANTA_2.8.0
## [16] igraph_1.0.1         formatR_1.2.1        knitr_1.11
##
## loaded via a namespace (and not attached):
##  [1] magrittr_1.5    splines_3.2.2   lattice_0.20-33 tools_3.2.2
##  [5] grid_3.2.2      snow_0.3-13     htmltools_0.2.6 survival_2.38-3
##  [9] digest_0.6.8    Matrix_1.2-2    evaluate_0.8    rmarkdown_0.8.1
## [13] stringi_1.0-1   expm_0.999-0    mvtnorm_1.0-3
```