Bioconductor's SPIA package

Adi L. Tarca[1,2,3], Purvesh Khatri[1] and Sorin Draghici[1]

March 25, 2009

[1]Department of Computer Science, Wayne State University
[2]Bioinformatics and Computational Biology Unit of the NIH Perinatology Research Branch
[3]Center for Molecular Medicine and Genetics, Wayne State University

1 Overview

This package implements the Signaling Pathway Impact Analysis (SPIA) algorithm described in Tarca et al. (2009), Khatri et al. (2007) and Draghici et al. (2007). SPIA uses the information from a set of differentially expressed genes and their fold changes, as well as pathways topology in order to assess the significance of the pathways in the condition under the study. The current version of SPIA algorithm uses KEGG signaling pathway data. SPIA ready KEGG pathway data for homo sapiens is included in the package, while for other popular organisms it can be automatically downloaded from
`http://bioinformaticsprb.med.wayne.edu/SPIA/build012309/` using package functionality.
The pathways included for each organism are those containing only directed relations between genes/proteins and no reactions.

2 Pathway analysis with SPIA package

This document provides basic introduction on how to use the `SPIA` package. For extended description of the methods used by this package please consult these references: Tarca et al. (2009); Khatri et al. (2007); Draghici et al. (2007).

We demonstrate the functionality of this package using a colorectal cancer dataset obtained using Affymetrix GeneChip technology and available through GEO (GSE4107). The experiment contains 10 normal samples and 12 colorectal cancer samples and is described by Hong et al. (2007). RMA preprocessing of the raw data was performed using the `affy` package, and a two group moderated t-test was applied using the `limma` package. The data frame obtained as an end result from the function topTable in limma is used as starting point for preparing the input data for SPIA. This data frame called `top` was made available in the `colorectalcancer` dataset included in the SPIA package:

```
> library(SPIA)
> data(colorectalcancer)
> options(digits = 3)
> head(top)
```

```
               ID logFC AveExpr    t  P.Value adj.P.Val    B
10738   201289_at  5.96    6.23 23.9 1.79e-17  9.78e-13 25.4
18604   209189_at  5.14    7.49 17.4 1.56e-14  2.84e-10 21.0
11143 201694_s_at  4.15    7.04 16.5 5.15e-14  7.04e-10 20.1
10490 201041_s_at  2.43    9.59 14.1 1.29e-12  1.41e-08 17.7
10913 201464_x_at  1.53    8.22 11.0 1.69e-10  1.15e-06 13.6
11463   202014_at  1.43    5.33 10.5 4.27e-10  2.42e-06 12.8
```

For SPIA to work, we need a vector with log2 fold changes between the two groups for all the genes considered to be differentially expressed. The names of this vector must be Entrez gene IDs. The following lines will add one additional column in the `top` data frame annotating each affymetrix probeset to an Etrez ID. Since there may be several probesets for the same Entrez ID, there are two easy ways to obtain one log fold change per gene. The first option is to use the fold change of the most significant probeset for each gene, while the second option is to average the log fold-changes of all probestes of the same gene. In the example below we used the former approach. The genes in this example are called differentially expressed provided that their FDR p-value is less than 0.05. The following lines start with the `top` data frame and produce two vectors that are required as input by `spia` function:

```
> library(hgu133plus2.db)
> x <- hgu133plus2ENTREZID
> top$ENTREZ <- unlist(as.list(x[top$ID]))
> top <- top[!is.na(top$ENTREZ), ]
> top <- top[!duplicated(top$ENTREZ), ]
> tg1 <- top[top$adj.P.Val < 0.05, ]
> DE_Colorectal = tg1$logFC
> names(DE_Colorectal) <- as.vector(tg1$ENTREZ)
> ALL_Colorectal = top$ENTREZ
```

The `DE_Colorectal` is a vector containing the log2 fold changes of the genes found to be differentially expressed between cancer and normal samples, and `ALL_Colorectal` is a vector with the Entrez IDs of all genes profiled on the microarray. The names of the `DE_Colorectal` are the Entrez gene IDs corresponding to the computed log fold-changes.

```
> DE_Colorectal[1:10]
```

```
 3491  2353  1958  1843  3725 23645  9510 84869  7432  1490
 5.96  5.14  4.15  2.43  1.53  1.43  3.94 -1.15  4.72  3.45
```

```
> ALL_Colorectal[1:10]
```

```
 [1] "3491"  "2353"  "1958"  "1843"  "3725"  "23645" "9510"  "84869" "7432"  "1490"
```

The SPIA algorithm takes as input the two vectors above and produces a table of pathways ranked from the most to the least significant. This can be achieved by calling the `spia` function as follows:

```
> res = spia(de = DE_Colorectal, all = ALL_Colorectal, organism = "hsa", nB = 2000, plots = FAl
+     beta = NULL)

Done pathway 1 : MAPK signaling pathway..
Done pathway 2 : ErbB signaling pathway..
Done pathway 3 : Calcium signaling pathway..
Done pathway 4 : Cytokine-cytokine recepto..
Done pathway 5 : Neuroactive ligand-recept..
Done pathway 6 : Cell cycle..
Done pathway 7 : p53 signaling pathway..
Done pathway 8 : Regulation of autophagy..
Done pathway 9 : mTOR signaling pathway..
Done pathway 10 : Apoptosis..
Done pathway 11 : Wnt signaling pathway..
Done pathway 12 : Notch signaling pathway..
Done pathway 13 : Hedgehog signaling pathwa..
Done pathway 14 : TGF-beta signaling pathwa..
Done pathway 15 : Axon guidance..
Done pathway 16 : VEGF signaling pathway..
Done pathway 17 : Focal adhesion..
Done pathway 18 : ECM-receptor interaction..
Done pathway 19 : Cell adhesion molecules (..
Done pathway 20 : Adherens junction..
Done pathway 21 : Tight junction..
Done pathway 22 : Gap junction..
Done pathway 23 : Complement and coagulatio..
Done pathway 24 : Antigen processing and pr..
Done pathway 25 : Toll-like receptor signal..
Done pathway 26 : Jak-STAT signaling pathwa..
Done pathway 27 : Natural killer cell media..
Done pathway 28 : T cell receptor signaling..
Done pathway 29 : B cell receptor signaling..
Done pathway 30 : Fc epsilon RI signaling p..
Done pathway 31 : Leukocyte transendothelia..
Done pathway 32 : Circadian rhythm..
Done pathway 33 : Long-term potentiation..
Done pathway 34 : Long-term depression..
Done pathway 35 : Olfactory transduction..
Done pathway 36 : Taste transduction..
Done pathway 37 : Regulation of actin cytos..
Done pathway 38 : Insulin signaling pathway..
Done pathway 39 : GnRH signaling pathway..
Done pathway 40 : Melanogenesis..
```

```
Done pathway 41 : Adipocytokine signaling p..
Done pathway 42 : Type II diabetes mellitus..
Done pathway 43 : Type I diabetes mellitus..
Done pathway 44 : Maturity onset diabetes o..
Done pathway 45 : Parkinson's disease..
Done pathway 46 : Amyotrophic lateral scler..
Done pathway 47 : Huntington's disease..
Done pathway 48 : Dentatorubropallidoluysia..
Done pathway 49 : Prion diseases..
Done pathway 50 : Vibrio cholerae infection..
Done pathway 51 : Epithelial cell signaling..
Done pathway 52 : Pathogenic Escherichia co..
Done pathway 53 : Colorectal cancer..
Done pathway 54 : Renal cell carcinoma..
Done pathway 55 : Pancreatic cancer..
Done pathway 56 : Endometrial cancer..
Done pathway 57 : Glioma..
Done pathway 58 : Prostate cancer..
Done pathway 59 : Thyroid cancer..
Done pathway 60 : Basal cell carcinoma..
Done pathway 61 : Melanoma..
Done pathway 62 : Bladder cancer..
Done pathway 63 : Chronic myeloid leukemia..
Done pathway 64 : Acute myeloid leukemia..
Done pathway 65 : Small cell lung cancer..
Done pathway 66 : Non-small cell lung cance..
Done pathway 67 : Asthma..
Done pathway 68 : Autoimmune thyroid diseas..
Done pathway 69 : Systemic lupus erythemato..
Done pathway 70 : Allograft rejection..
Done pathway 71 : Graft-versus-host disease..
```

```
> res$Name = substr(res$Name, 1, 10)
> res[1:15, ]
```

```
         Name    ID pSize NDE     tA     pNDE    pPERT       pG    pGFdr   pGFWER    Status
1  Parkinson' 05012   109  59 -12.10 1.49e-15 0.040000 2.29e-15 1.58e-13 1.58e-13 Inhibited
2  Focal adhe 04510   181  65  63.27 2.54e-07 0.000005 3.60e-11 1.24e-09 2.48e-09 Activated
3  ECM-recept 04512    76  26  19.93 2.14e-03 0.000005 2.07e-07 4.75e-06 1.43e-05 Activated
4  Axon guida 04360   119  47   8.06 5.03e-07 0.425000 3.50e-06 6.03e-05 2.41e-04 Activated
5  Colorectal 05210    77  26   7.31 2.63e-03 0.051000 1.33e-03 1.84e-02 9.18e-02 Activated
6  MAPK signa 04010   250  72   3.67 3.40e-04 0.602000 1.94e-03 2.23e-02 1.34e-01 Activated
7  Wnt signal 04310   139  42  -8.52 2.03e-03 0.202000 3.61e-03 3.19e-02 2.49e-01 Inhibited
8  Regulation 04810   197  57   8.06 1.18e-03 0.357000 3.70e-03 3.19e-02 2.55e-01 Activated
9  Renal cell 05211    63  21 -13.96 7.63e-03 0.072000 4.67e-03 3.58e-02 3.23e-01 Inhibited
10 Notch sign 04330    45  17   3.99 3.81e-03 0.470000 1.31e-02 9.04e-02 9.04e-01 Activated
```

```
11 Dentatorub 05050    14   8   0.00 2.18e-03 1.000000 1.55e-02 9.75e-02 1.00e+00 Inhibited
12 Circadian  04710     9   6   0.00 2.84e-03 1.000000 1.95e-02 9.91e-02 1.00e+00 Inhibited
13  Apoptosis 04210    86  24 -15.28 4.22e-02 0.068000 1.97e-02 9.91e-02 1.00e+00 Inhibited
14 Pathogenic 05130    46  13  17.40 1.05e-01 0.028000 2.01e-02 9.91e-02 1.00e+00 Activated
15 Tight junc 04530   123  36   2.55 7.11e-03 0.499000 2.36e-02 1.08e-01 1.00e+00 Activated
```

If the `plots` argument is set to `TRUE` in the function call above, a plot like the one shown in Figure 1 is produced for each pathway on which there are differentially expressed genes. These plots are saved in a pdf file in the current directory.
An overall picture of the pathways significance according to both the over-representation evidence and perturbations based evidence can be obtained with the function `plotP` and shown in Figure 2. In this plot, the horizontal axis represents the p-value (minus log of) corresponding to the probability of obtaining at least the observed number of genes (NDE) on the given pathway just by chance. The vertical axis represents the p-value (minus log of) corresponding to the probability of obtaining the observed total accumulation (tA) or more extreme on the given pathway just by chance. The computation of pPERT is described in Tarca et al. (2009). In Figure 2 each pathway is shown as a bullet point, and those significant at 5% (set by the `threshold` argument in `plotP`) after Bonferroni correction are shown in red.

SPIA algorithm is illustrated also using the Vessels dataset:

```
> data(Vessels)
> res <- spia(de = DE_Vessels, all = ALL_Vessels, organism = "hsa", nB = 500, plots = FALSE,
+     beta = NULL, verbose = FALSE)
> res$Name = substr(res$Name, 1, 10)
> res[1:15, ]
```

```
         Name    ID pSize NDE     tA     pNDE pPERT       pG  pGFdr pGFWER    Status
1  Axon guida 04360   128  12 -5.794 0.000208 0.116 0.000281 0.0132 0.0171 Inhibited
2  Focal adhe 04510   199  16 -5.384 0.000123 0.316 0.000434 0.0132 0.0265 Inhibited
3  Neuroactiv 04080   255  18 -0.510 0.000247 0.512 0.001260 0.0256 0.0768 Inhibited
4  Notch sign 04330    45   4  6.143 0.034132 0.008 0.002514 0.0383 0.1533 Activated
5  Complement 04610    67   7  3.967 0.002325 0.176 0.003602 0.0439 0.2197 Activated
6  Wnt signal 04310   149  12  1.252 0.000830 0.608 0.004337 0.0441 0.2645 Activated
7  Graft-vers 05332    40   6  0.000 0.000710 1.000 0.005859 0.0498 0.3574 Inhibited
8  Type I dia 04940    42   6  0.000 0.000927 1.000 0.007400 0.0498 0.4514 Inhibited
9  Regulation 04810   210  14  2.416 0.002001 0.492 0.007799 0.0498 0.4757 Activated
10     Asthma 05310    29   5  0.000 0.001038 1.000 0.008167 0.0498 0.4982 Inhibited
11 Antigen pr 04612    86   7  1.513 0.009235 0.152 0.010624 0.0589 0.6481 Activated
12 Allograft  05330    36   5  0.000 0.002815 1.000 0.019346 0.0983 1.0000 Inhibited
13 Leukocyte  04670   116   9 -0.804 0.004608 0.808 0.024547 0.1152 1.0000 Inhibited
14 TGF-beta s 04350    88   7 -1.766 0.010418 0.440 0.029268 0.1275 1.0000 Inhibited
15 Cytokine-c 04060   258  12 -2.527 0.052351 0.100 0.032732 0.1275 1.0000 Inhibited
```

Note that the results for these datasets my differ from the ones described in Tarca et al. (2009) since a) the pathways database used herein was updated and b) the default beta values were changed.

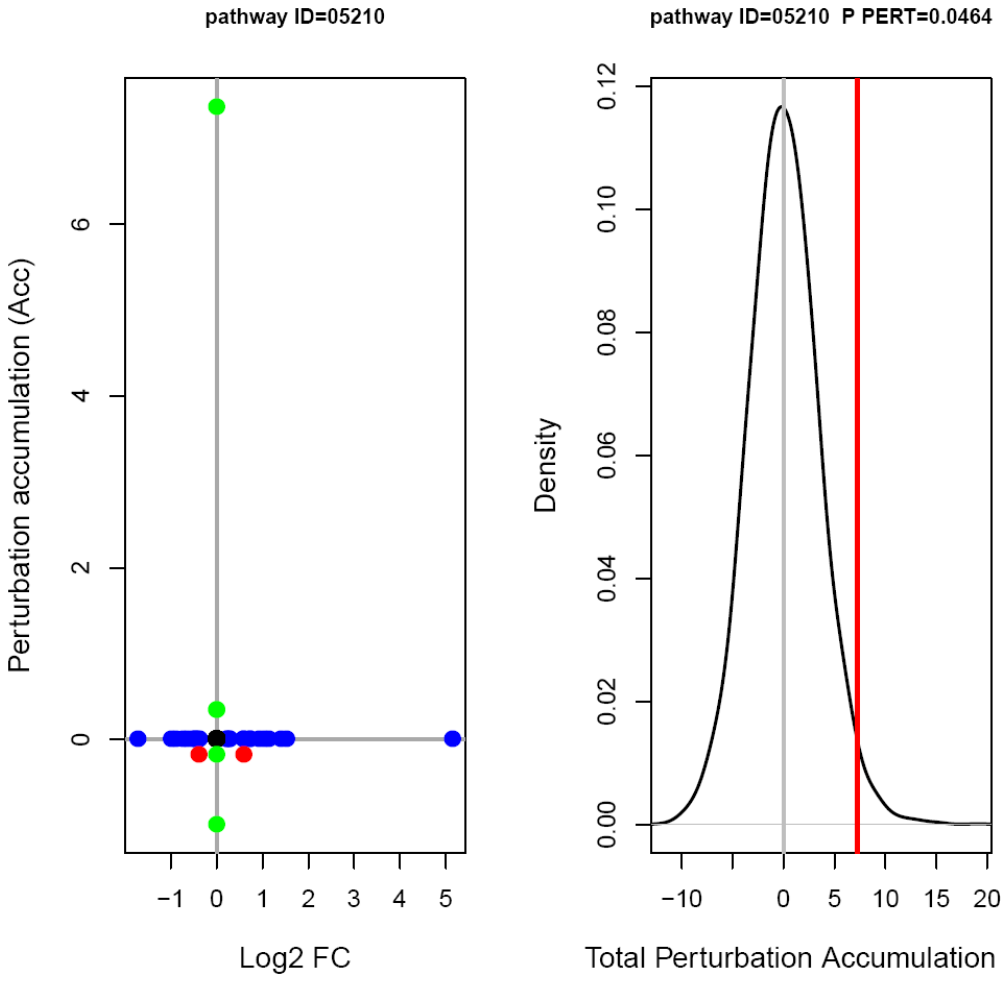

Figure 1: Perturbations plot for colorectal cancer pathway (KEGG ID hsa:05210) using the `colorectalcancer` dataset. The perturbation of all genes in the pathway are shown as a function of their initial log2 fold changes (left panel). Non DE genes are assigned 0 log2 fold-change. The null distribution of the net accumulated perturbations is also given (right panel). The observed net accumulation tA with the real data is shown as a red vertical line.

```
> plotP(res, threshold = 0.05)
```

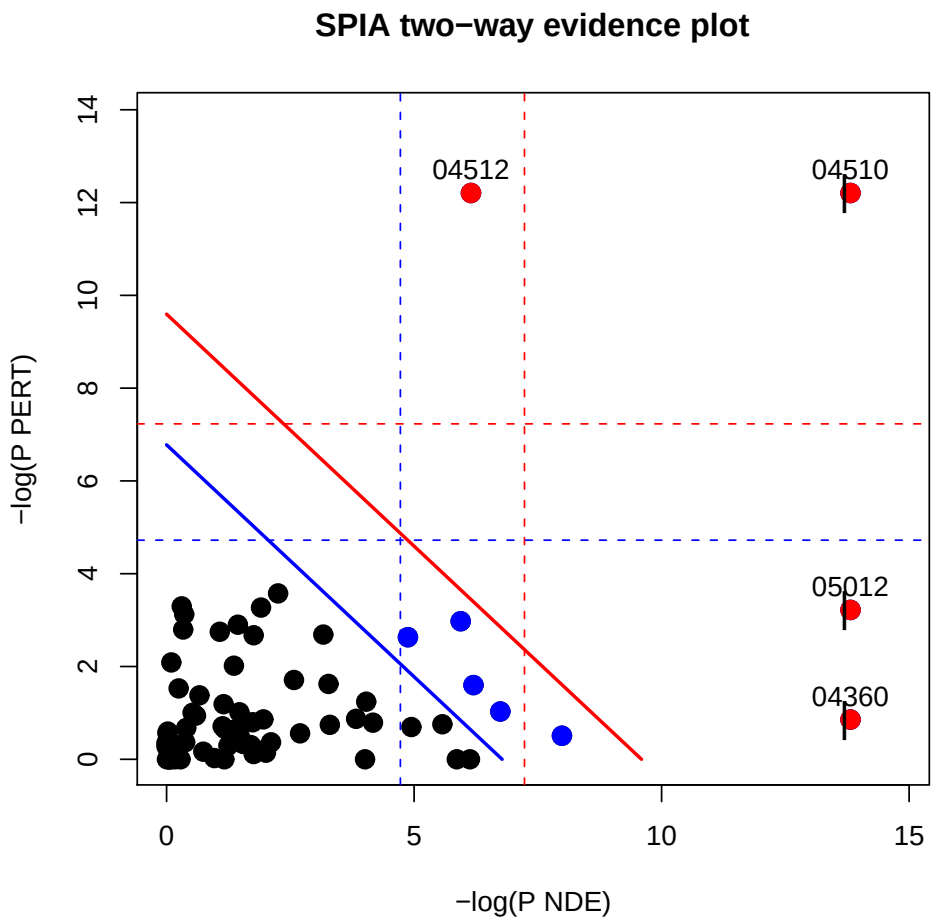

Figure 2: SPIA evidence plot for the colorectal cancer dataset. Each pathway is represented by one dot. The pathways at the right of the red oblique line are significant after Bonferroni correction of the global p-values, pG. The pathways at the right of the blue oblique line are significant after a FDR correction of the global p-values, pG.

The directed adjacency matrices of the graphs describing the different types of relations between genes/proteins (such as activation or repression) used by SPIA are available in the `extdata/hsaSPIA.RData` file for the homo sapiens organism. The types of relations considered by SPIA and the default weight (beta coefficient) given to them are:

```
> rel <- c("activation", "compound", "binding/association", "expression", "inhibition", "activa
+     "phosphorylation", "indirect", "inhibition_phosphorylation", "dephosphorylation_inhibitio
+     "dissociation", "dephosphorylation", "activation_dephosphorylation", "state", "activation
+     "inhibition_ubiquination", "ubiquination", "expression_indirect", "indirect_inhibition",
+     "repression", "binding/association_phosphorylation", "dissociation_phosphorylation",
+     "indirect_phosphorylation")
> beta = c(1, 0, 0, 1, -1, 1, 0, 0, -1, -1, 0, 0, 1, 0, 1, -1, 0, 1, -1, -1, 0, 0, 0)
> names(beta) <- rel
> cbind(beta)
```

```
                                   beta
activation                            1
compound                              0
binding/association                   0
expression                            1
inhibition                           -1
activation_phosphorylation            1
phosphorylation                       0
indirect                              0
inhibition_phosphorylation           -1
dephosphorylation_inhibition         -1
dissociation                          0
dephosphorylation                     0
activation_dephosphorylation          1
state                                 0
activation_indirect                   1
inhibition_ubiquination              -1
ubiquination                          0
expression_indirect                   1
indirect_inhibition                  -1
repression                           -1
binding/association_phosphorylation   0
dissociation_phosphorylation          0
indirect_phosphorylation              0
```

A 0 value for a given relation type results in discarding those type of relations from the analysis for all pathways. The default values of `beta` can changed by the user at any time by setting the `beta` argument of the `spia` function call.
Other organisms' KEGG pathway data can be made available to the `spia` function by using the function `getSPIAMatrices`. For example, the mus musculus pathway data can be made availabe using:

```
> getSPIAMatrices(organism = "mmu")
```

The execution of the function above will download the latest processed pathway data from `http://bioinformaticsprb.med.wayne.edu/SPIA` for the requested organism and creates a "[org]SPIA.RData" file (in this case org=mmu) in the `extdata` directory of the SPIA package, and therefore makes is available to the function `spia`.
The user has the ability to generate his own gene/protein relation data and put it in a list format as the one shown in the hsaSPIA.RData file. In this file, each pathway data is included in a list:

```
> load(file = paste(system.file("extdata", package = "SPIA"), paste("/hsa", "SPIA", sep = ""),
+     ".RData", sep = ""))
> names(path.info[["05210"]])

 [1] "activation"                          "compound"
 [3] "binding/association"                 "expression"
 [5] "inhibition"                          "activation_phosphorylation"
 [7] "phosphorylation"                     "indirect"
 [9] "inhibition_phosphorylation"          "dephosphorylation_inhibition"
[11] "dissociation"                        "dephosphorylation"
[13] "activation_dephosphorylation"        "state"
[15] "activation_indirect"                 "inhibition_ubiquination"
[17] "ubiquination"                        "expression_indirect"
[19] "indirect_inhibition"                 "repression"
[21] "binding/association_phosphorylation" "dissociation_phosphorylation"
[23] "indirect_phosphorylation"            "nodes"
[25] "title"                               "NumberOfReactions"

> path.info[["05210"]][["activation"]][48:60, 55:60]

     8313 5900 5879 5880 5881 332
369     0    0    0    0    0   0
5894    0    0    0    0    0   0
673     0    0    0    0    0   0
5599    0    0    1    1    1   0
5601    0    0    1    1    1   0
5602    0    0    1    1    1   0
8312    0    0    0    0    0   0
8313    0    0    0    0    0   0
5900    0    0    0    0    0   0
5879    0    1    0    0    0   0
5880    0    1    0    0    0   0
5881    0    1    0    0    0   0
332     0    0    0    0    0   0
```

In the matrix above, only 0 and 1 values are allowed. 1 means the gene/protein given by the column has a relation of type "activation" with the gene/protein given by the row of the matrix.

Using other R packages such as `graph` and `Rgraphviz` one can visualize the richness of gene/protein relations of each type in each pathway. Firstly we load the required packages and create a function that can be used to plot as a graph each type of relation of any pathway, as used by SPIA.

```
> library(graph)
> library(Rgraphviz)
> plotG <- function(B) {
+     nnms <- NULL
+     colls <- NULL
+     mynodes <- colnames(B)
+     L <- list()
+     n <- dim(B)[1]
+     for (i in 1:n) {
+         L[i] <- list(edges = rownames(B)[abs(B[, i]) > 0])
+         if (sum(B[, i] != 0) > 0) {
+             nnms <- c(nnms, paste(colnames(B)[i], rownames(B)[B[, i] != 0], sep = "~"))
+         }
+     }
+     names(L) <- rownames(B)
+     g <- new("graphNEL", nodes = mynodes, edgeL = L, edgemode = "directed")
+     plot(g)
+ }
```

We plot then the "activation" relations in the ErbB signaling pathway, based on the `hsaSPIA` data. For more details on how to use the main function in this package use "?spia".

References

S. Draghici, P. Khatri, A. Tarca, K. Amin, A. Done, C. Voichita, C. Georgescu, and R. Romero. A systems biology approach for pathway level analysis. *Genome Research*, 17, 2007.

Y. Hong, K. S. Ho, K. W. Eu, and P. Y. Cheah. A susceptibility gene set for early onset colorectal cancer that integrates diverse signaling pathways: implication for tumorigenesis. *Clin Cancer Res*, 13(4):1107–14, 2007.

P. Khatri, S. Draghici, A. L. Tarca, S. S. Hassan, and R. Romero. A system biology approach for the steady-state analysis of gene signaling networks. In *12th Iberoamerican Congress on Pattern Recognition*, Valparaiso, Chile, November 13-16 2007.

A. L. Tarca, S. Draghici, P. Khatri, S. Hassan, P. Mital, J. Kim, C. Kim, J. P. Kusanovic, and R. Romero. A signaling pathway impact analysis for microarray experiments. *Bioinformatics*, 25:75–82, 2009.

```
> plotG(path.info[["04012"]][["activation"]])
```

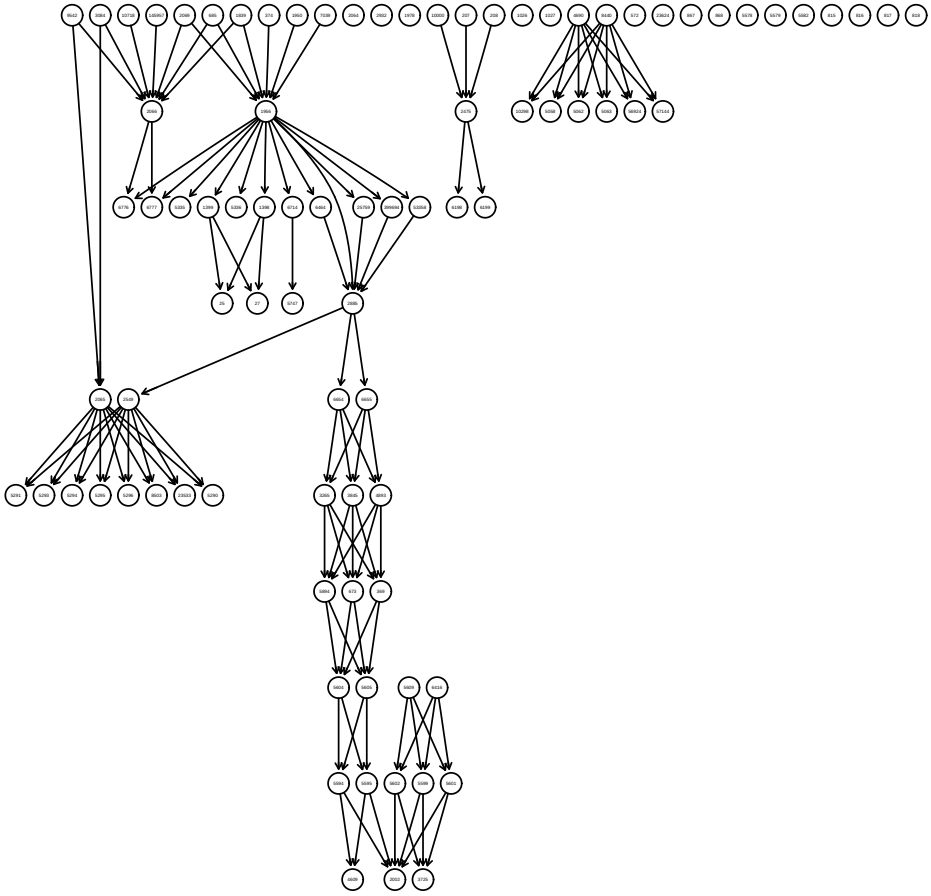

Figure 3: Display of the "activation" relations in the ErbB signaling pathway, based on the hsaSPIA data.