

Bioconductor's SPIA package

Adi L. Tarca^{1,2,3}, Purvesh Khatri¹ and Sorin Draghici¹

October 17, 2016

¹Department of Computer Science, Wayne State University

²Bioinformatics and Computational Biology Unit of the NIH Perinatology Research Branch

³Center for Molecular Medicine and Genetics, Wayne State University

1 Overview

This package implements the Signaling Pathway Impact Analysis (SPIA) algorithm described in [? , ? and ?](#). 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 The current version of SPIA algorithm includes out-of-date KEGG signaling pathway data for hsa and mmu organisms for illustration purposes. However, the current version of the package includes functionality to generate the required up-to-date processed pathway data from KEGG xml (KGML) files that licensed users can download for the organism of interest from KEGG's ftp site. Also, these files can be downloaded individually using the Download KEGML button from each pathway's web page. The pathways that will be processed and analyzed for a given organism are those i) containing at least one relation between genes/proteins considered by SPIA, and ii) having no reactions.

The outdated KEGG data that was preprocessed for SPIA analysis and is included for the hsa and mmu organisms was downloaded from KEGG's website on: 09/07/2012. For a list of changes in SPIA compared to previous versions see the last section in this document.

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: [??? .](#)

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 [? .](#) 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	ENTREZ
10738	201289_at	5.96	6.23	23.9	1.79e-17	9.78e-13	25.4	3491
18604	209189_at	5.14	7.49	17.4	1.56e-14	2.84e-10	21.0	2353
11143	201694_s_at	4.15	7.04	16.5	5.15e-14	7.04e-10	20.1	1958
10490	201041_s_at	2.43	9.59	14.1	1.29e-12	1.41e-08	17.7	1843
10913	201464_x_at	1.53	8.22	11.0	1.69e-10	1.15e-06	13.6	3725
11463	202014_at	1.43	5.33	10.5	4.27e-10	2.42e-06	12.8	23645

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 Entrez 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 probesets 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 adjusted p-values (q-values) are 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$adj.P.Val<0.1,]
> DE_Colorectal=tg1$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"
[10] "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:

```
> # pathway analysis based on combined evidence; # use nB=2000 or more for more accurate results
> res=spia(de=DE_Colorectal,all=ALL_Colorectal,organism="hsa",nB=2000,plots=FALSE,beta=NULL,cor=0)
> #make the output fit this screen
> res$Name=substr(res$Name,1,10)
> #show first 15 pathways, omit KEGG links
> res[1:20,-12]
```

	Name	ID	pSize	NDE	pNDE	tA	pPERT	pG	pGFdr
1	Focal adhe	04510	174	87	9.98e-08	101.78	0.000005	1.46e-11	2.01e-09
2	Alzheimer'	05010	144	83	2.47e-11	-5.79	0.213000	1.42e-10	9.73e-09
3	ECM-recept	04512	72	41	4.03e-06	26.06	0.000005	5.17e-10	2.36e-08
4	Parkinson'	05012	105	63	6.37e-10	-11.45	0.072000	1.14e-09	3.90e-08
5	Pathways i	05200	294	123	4.15e-05	67.88	0.006000	4.03e-06	1.11e-04
6	PPAR signa	03320	64	37	7.31e-06	-3.02	0.049000	5.67e-06	1.29e-04
7	Axon guida	04360	117	58	1.78e-05	12.93	0.127000	3.17e-05	6.19e-04
8	Small cell	05222	73	33	6.95e-03	26.79	0.001000	8.95e-05	1.38e-03
9	MAPK signa	04010	239	103	4.16e-05	11.00	0.186000	9.87e-05	1.38e-03
10	Huntington	05016	163	75	3.36e-05	-3.12	0.235000	1.01e-04	1.38e-03
11	Fc gamma R	04666	80	42	4.55e-05	-11.89	0.219000	1.25e-04	1.55e-03
12	Regulation	04810	189	83	1.03e-04	10.74	0.407000	4.63e-04	5.28e-03
13	Glutamater	04724	113	45	2.69e-02	-11.55	0.007000	1.80e-03	1.90e-02
14	Pathogenic	05130	44	21	1.41e-02	17.48	0.020000	2.58e-03	2.42e-02
15	Bacterial	05100	61	32	3.61e-04	3.16	0.802000	2.65e-03	2.42e-02
16	Wnt signal	04310	136	58	2.42e-03	-6.90	0.326000	6.44e-03	5.51e-02
17	Renal cell	05211	61	28	9.65e-03	-8.25	0.097000	7.46e-03	6.01e-02
18	Transcript	05202	145	59	7.64e-03	-1.18	0.145000	8.65e-03	6.58e-02
19	B cell rec	04662	70	32	6.37e-03	-10.17	0.196000	9.59e-03	6.76e-02
20	ErbB signa	04012	76	34	7.54e-03	-17.50	0.171000	9.87e-03	6.76e-02

	pGFWER	Status
1	2.01e-09	Activated
2	1.95e-08	Inhibited
3	7.08e-08	Activated
4	1.56e-07	Inhibited
5	5.53e-04	Activated
6	7.77e-04	Inhibited
7	4.34e-03	Activated
8	1.23e-02	Activated
9	1.35e-02	Activated
10	1.38e-02	Inhibited
11	1.71e-02	Inhibited
12	6.34e-02	Activated
13	2.47e-01	Inhibited
14	3.54e-01	Activated

```

15 3.63e-01 Activated
16 8.82e-01 Inhibited
17 1.00e+00 Inhibited
18 1.00e+00 Inhibited
19 1.00e+00 Inhibited
20 1.00e+00 Inhibited

```

If the `plots` argument is set to `TRUE` in the function call above, a plot like the one shown in Figure ?? 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 ?. The Colorectal cancer pathway is shown in green.

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 ?. In Figure ?? 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.

The default method to combine pPERT and pNDE is Fisher's product method, as was described in ?.

Alternatively, the two types of evidence can be combined using a normal inversion method which gives smaller pG values when pPERT and pNDE are low simultaneously. This is in contrast with Fisher's method that may yield small pG values when only one of the two p-values is low. To use the normal inversion method, one can set the argument `combine="norminv"` when the `spia` function is called, or by recomputing pG values starting with a result data frame produced by `spia` function. This latter approach is illustrated below where a call is made to the function `combfunc`. SPIA algorithm is illustrated also using the Vessels dataset:

```

> data(Vessels)
> # pathway analysis based on combined evidence; # use nB=2000 or more for more accurate results
> res<-spia(de=DE_Vessels,all=ALL_Vessels,organism="hsa",nB=500,plots=FALSE,beta=NULL,verbose=1)
> #make the output fit this screen
> res$Name=substr(res$Name,1,10)
> #show first 15 pathways, omit KEGG links
> res[1:15,-12]

```

	Name	ID	pSize	NDE	pNDE	tA	pPERT	pG	pGFdr	pGFWER
1	Axon guida	04360	128	12	2.08e-04	-6.4917	0.040	0.000106	0.0132	0.0132
2	Staphyloco	05150	51	8	6.74e-05	1.9922	0.440	0.000339	0.0193	0.0424
3	Focal adhe	04510	200	16	1.31e-04	-6.3721	0.320	0.000463	0.0193	0.0579
4	Regulation	04810	209	15	6.66e-04	7.3879	0.128	0.000884	0.0276	0.1105
5	Viral myoc	05416	69	8	5.76e-04	-1.6656	0.272	0.001529	0.0322	0.1912
6	Rheumatoid	05323	89	10	1.59e-04	0.0000	1.000	0.001547	0.0322	0.1933

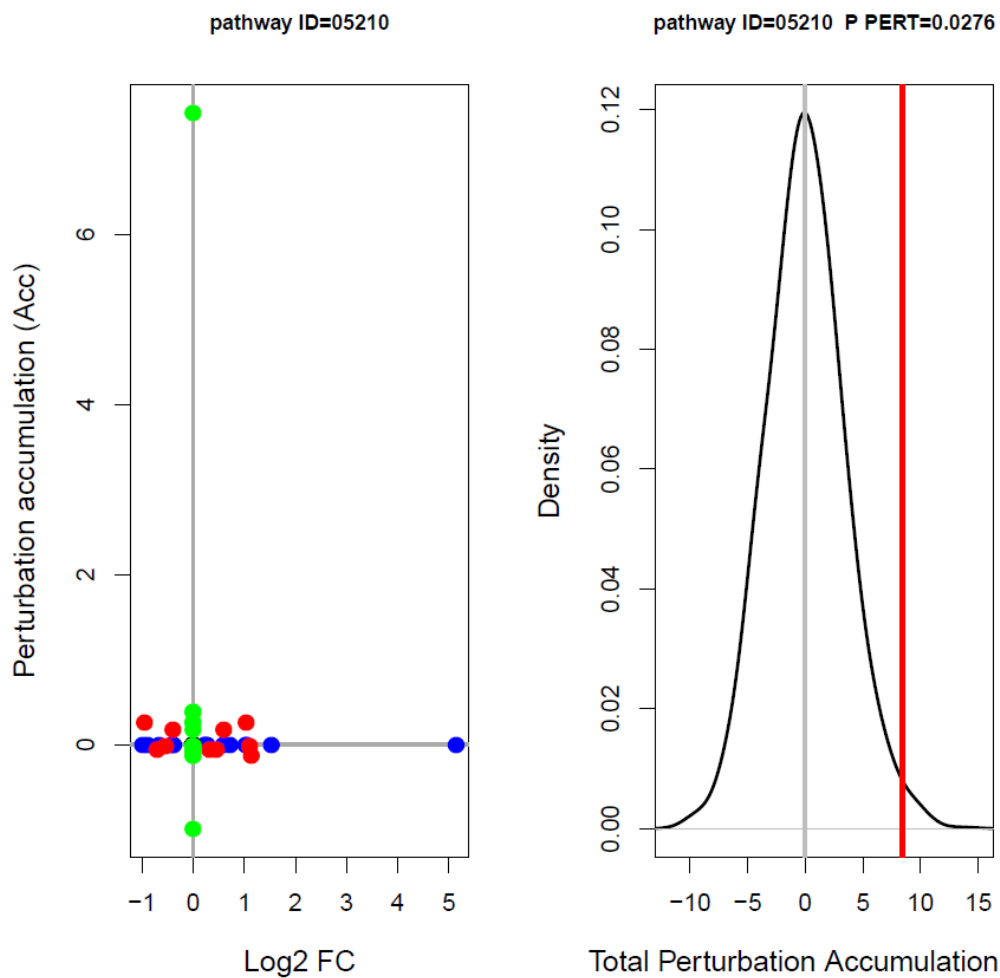


Figure 1: Perturbations plot for colorectal cancer pathway (KEGG ID hsa:05210) using the `colorectal_cancer` dataset. The perturbation of all genes in the pathway are shown as a function of their initial log2 fold changes (left panel). Non DE genes are assigned 0 log2 fold-change. The null distribution 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)
> points(I(-log(pPERT))~I(-log(pNDE)), data=res[res$ID=="05210",], col="green", pch=19, cex=1.5)
>

```

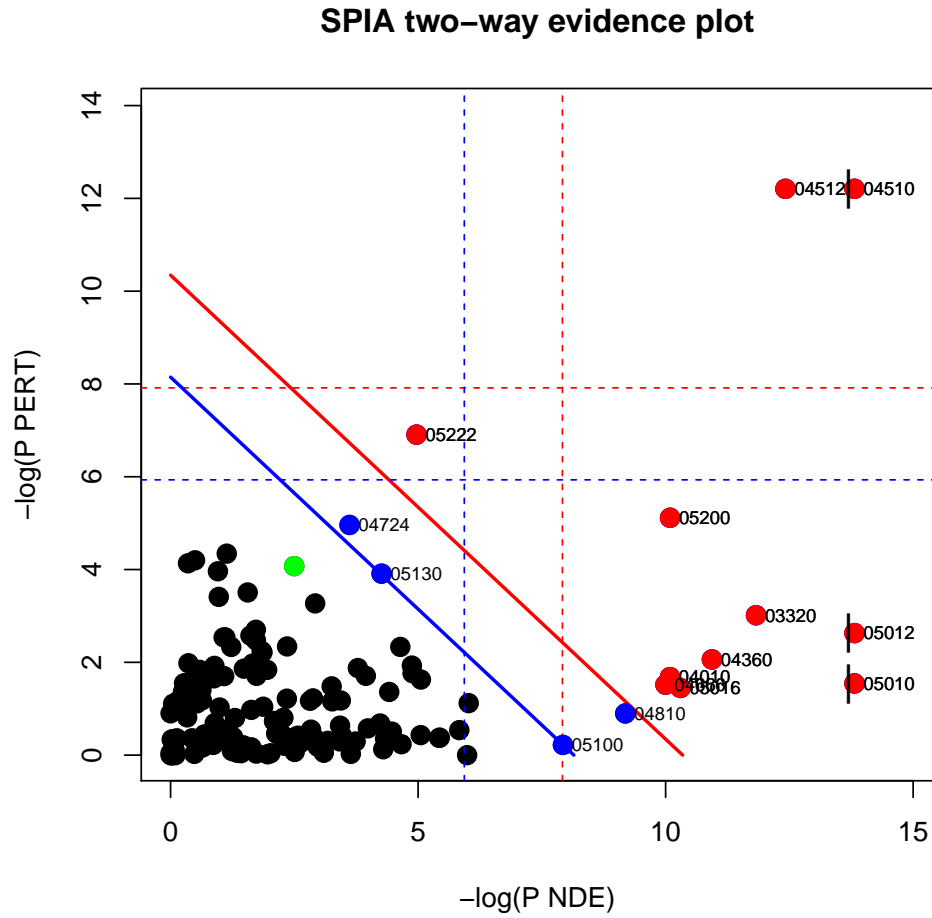


Figure 2: SPIA evidence plot for the colorectal cancer dataset. Each pathway is represented by one dot. The pathways at the right of the red oblique line are significant after Bonferroni correction of the global p-values, pG, obtained by combining the pPERT and pNDE using Fisher's method. The pathways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

```

> res$pG=combfunc(res$pNDE,res$pPERT,combine="norminv")
> res$pGFdr=p.adjust(res$pG,"fdr")
> res$pGFWER=p.adjust(res$pG,"bonferroni")
> plotP(res,threshold=0.05)
> points(I(-log(pPERT))~I(-log(pNDE)),data=res[res$ID=="05210",],col="green",pch=19,cex=1.5)
>

```

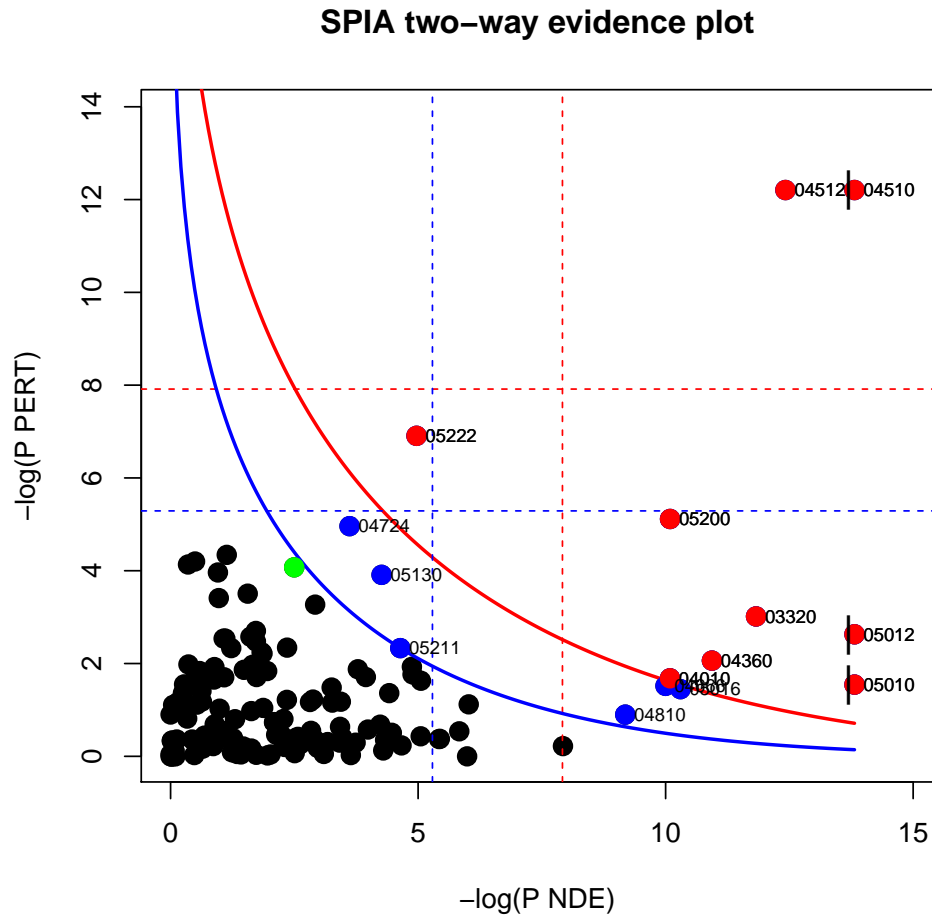


Figure 3: SPIA evidence plot for the colorectal cancer dataset. Each pathway is represented by one dot. The pathways at the right of the red curve are significant after Bonferroni correction of the global p-values, pG , obtained by combining the $pPERT$ and $pNDE$ using the normal inversion method. The pathways at the right of the blue curve line are significant after a FDR correction of the global p-values, pG .

7	Intestinal	04672	46	7	2.34e-04	0.0000	1.000	0.002193	0.0323	0.2741
8	Neuroactiv	04080	272	18	5.35e-04	-0.5104	0.480	0.002380	0.0323	0.2975
9	HTLV-I inf	05166	258	18	2.84e-04	0.4030	0.940	0.002466	0.0323	0.3083
10	Antigen pr	04612	75	7	4.40e-03	1.4922	0.064	0.002586	0.0323	0.3232
11	Leishmania	05140	68	8	5.22e-04	0.0896	0.972	0.004353	0.0495	0.5441
12	Graft-vers	05332	40	6	7.10e-04	0.0000	1.000	0.005859	0.0579	0.7324
13	Notch sign	04330	46	4	3.66e-02	6.9459	0.020	0.006017	0.0579	0.7522
14	Complement	04610	67	7	2.33e-03	4.9703	0.364	0.006834	0.0610	0.8542
15	Type I dia	04940	42	6	9.27e-04	0.0000	1.000	0.007400	0.0617	0.9250

Status

- 1 Inhibited
- 2 Activated
- 3 Inhibited
- 4 Activated
- 5 Inhibited
- 6 Inhibited
- 7 Inhibited
- 8 Inhibited
- 9 Activated
- 10 Activated
- 11 Activated
- 12 Inhibited
- 13 Activated
- 14 Activated
- 15 Inhibited

The pathway image as provided by KEGG having the differentially expressed genes highlighted in red can be obtained by pasting in a web browser the links available in the KEGGLINK column of the data frame produced by the function spia. For example,

```
> res[, "KEGGLINK"] [20]
```

```
[1] "http://www.genome.jp/dbget-bin/show_pathway?hsa05330+3108+3109+3111+3113+3122"
```

is the link that would display the image of the 20th pathway in the res dataframe above.

Note that the results for these datasets may differ from the ones described in ? since a) the pathways database used herein was updated and b) the default beta values were changed.

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",
+ "activation_phosphorylation", "phosphorylation", "inhibition_phosphorylation",
+ "inhibition_dephosphorylation", "dissociation", "dephosphorylation",
+ "activation_dephosphorylation", "state change", "activation_indirect effect",
```

```

+ "inhibition_ubiquination","ubiquination", "expression_indirect effect",
+ "inhibition_indirect effect","repression","dissociation_phosphorylation",
+ "indirect effect_phosphorylation","activation_binding/association",
+ "indirect effect","activation_compound","activation_ubiquination")
> beta=c(1,0,0,1,-1,1,0,-1,-1,0,0,1,0,1,-1,0,1,-1,-1,0,0,1,0,1,1)
> names(beta)<-rel
> cbind(beta)

```

	beta
activation	1
compound	0
binding/association	0
expression	1
inhibition	-1
activation_phosphorylation	1
phosphorylation	0
inhibition_phosphorylation	-1
inhibition_dephosphorylation	-1
dissociation	0
dephosphorylation	0
activation_dephosphorylation	1
state change	0
activation_indirect effect	1
inhibition_ubiquination	-1
ubiquination	0
expression_indirect effect	1
inhibition_indirect effect	-1
repression	-1
dissociation_phosphorylation	0
indirect effect_phosphorylation	0
activation_binding/association	1
indirect effect	0
activation_compound	1
activation_ubiquination	1

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 be changed by the user at any time by setting the **beta** argument of the **spia** function call.

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/hsaSPIA.RData",package="SPIA")))
> names(path.info[["05210"]])

[1] "activation"           "compound"
[3] "binding/association"  "expression"
[5] "inhibition"           "activation_phosphorylation"

```

```

[7] "phosphorylation"           "inhibition_phosphorylation"
[9] "inhibition_dephosphorylation" "dissociation"
[11] "dephosphorylation"         "activation_dephosphorylation"
[13] "state change"              "activation_indirect effect"
[15] "inhibition_ubiquination"    "ubiquination"
[17] "expression_indirect effect" "inhibition_indirect effect"
[19] "repression"                 "dissociation_phosphorylation"
[21] "indirect effect_phosphorylation" "activation_binding/association"
[23] "indirect effect"           "activation_compound"
[25] "activation_ubiquination"     "nodes"
[27] "title"                      "NumberOfReactions"

```

```
> path.info[["05210"]][["activation"]][25:35,30:40]
```

	5602	8312	8313	5900	387	5879	5880	5881	332	4609	595
369	0	0	0	0	0	0	0	0	0	0	0
5894	0	0	0	0	0	0	0	0	0	0	0
673	0	0	0	0	0	0	0	0	0	0	0
5599	0	0	0	0	1	1	1	1	0	0	0
5601	0	0	0	0	1	1	1	1	0	0	0
5602	0	0	0	0	1	1	1	1	0	0	0
8312	0	0	0	0	0	0	0	0	0	0	0
8313	0	0	0	0	0	0	0	0	0	0	0
5900	0	0	0	0	0	0	0	0	0	0	0
387	0	0	0	1	0	0	0	0	0	0	0
5879	0	0	0	1	0	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.

```

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

```

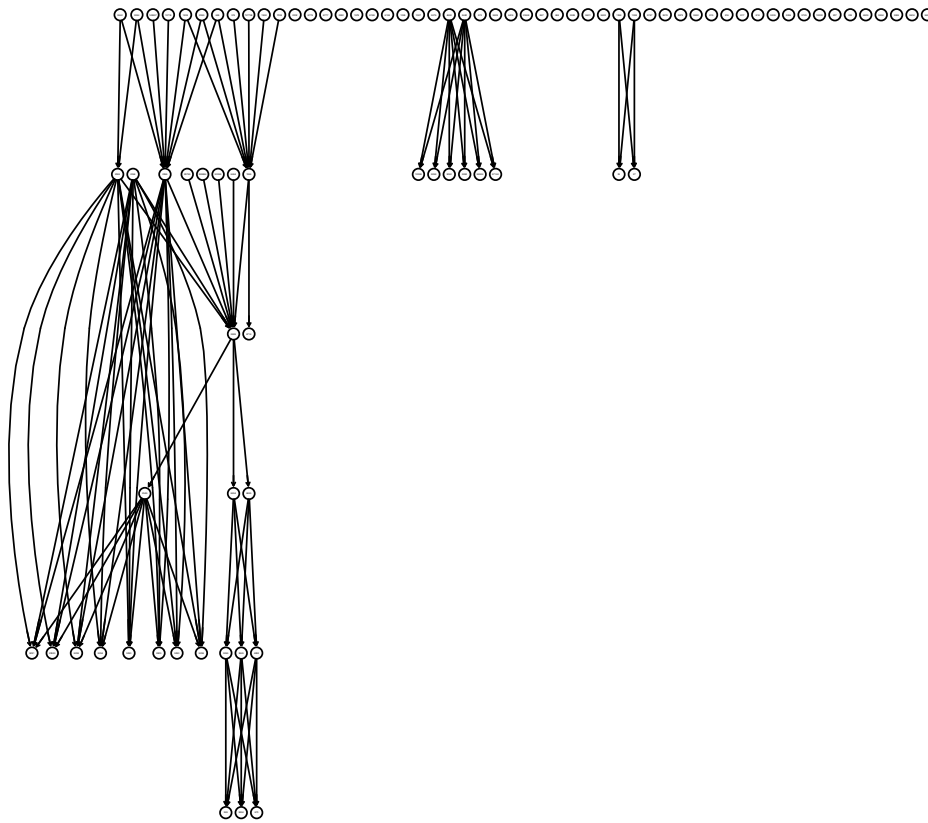


Figure 4: Display of the "activation" relations in the ErbB signaling pathway, based on the **hsaSPIA** data.

3 Parsing up-to-date KEGG xml files for use with SPIA

Here we assume that the user obtained the KGML (xml) files for all pathways of interest for a given organism from the KEGG ftp site (or downloaded them one by one from the KEGG web site). As an example we included four such files in the `extdata/keggxml/hsa` folder of the SPIA package installation to demonstrate how to parse these files and run SPIA on the resulting collection of pathways.

```
> mydir=system.file("extdata/keggxml/hsa",package="SPIA")
> dir(mydir)

[1] "hsa03013.xml" "hsa03050.xml" "hsa04914.xml" "hsa05210.xml"

> makeSPIAdata(kgml.path=mydir,organism="hsa",out.path=".")

[1] TRUE

> res<-spia(de=DE_Colorectal, all=ALL_Colorectal, organism="hsa",data.dir=".")

Done pathway 1 : RNA transport..
Done pathway 2 : Progesterone-mediated oocyte m..
Done pathway 3 : Colorectal cancer..

> res[, -12]
```

	Name	ID	pSize	NDE	pNDE	tA	pPERT
1	Colorectal cancer	05210	57	23	0.0823	8.49	0.027
2	Progesterone-mediated oocyte maturation	04914	76	29	0.1072	2.30	0.653
3	RNA transport	03013	133	30	0.9874	0.00	1.000

	pG	pGFdr	pGFWER	Status
1	0.0158	0.0474	0.0474	Activated
2	0.2561	0.3842	0.7683	Activated
3	0.9999	0.9999	1.0000	Inhibited

For more details on how to use the main function in this package use `?spia`.

A commercial version of SPIA called PathwayGuide that includes additional capabilities in terms of visualisation, speed and user interface is available from <http://www.advaitabio.com/>.

4 Changes in SPIA 2.10 vs 2.9

The current version (2.10) contains the following changes compared to the previous version (2.9): A function `makeSPIAdata` was added that generates `xxxSPIA.RData` files from KGML (xml) files provided by the user. The package will not contain anymore up-to-date KEGG pathway data since the access to the KEGG ftp server requires a license.