psygenet2r: An R package for querying PsyGeNET and to perform comorbidity studies in psychiatric disorders

Alba Gutierrez-Sacristan Carles Hernandez-Ferrer Juan R. Gonzalez
Laura I. Furlong

August 25, 2016

Contents

1 Introduction 2
1.1 Background . . . 2
1.2 Installation . . . 3
1.3 DataGeNET.Psy object . . . 3

2 PsyGeNET and psygenet2r 3

3 Retrieve gene-disease associations (GDAs) from psygenet2r 4
3.1 Using genes as a query . . . 4
3.1.1 Using as a query a single gene . . . 4
3.1.2 Ploting the results of a Single Gene Query . . . 5
3.1.3 Using as a query a list of genes . . . 7
3.1.4 Ploting the results of the query using a list of genes . . . 8
3.1.5 Ploting results of the Multiple Gene Query . . . 11
3.1.6 Enrichment analysis for a list of genes . . . 16
3.1.7 Enrichment analysis based on anatomical terms (TopAnat) for a list of genes 17
3.1.8 Sentences that report a GDA . . . 17
3.2 Using diseases as a query . . . 18
3.2.1 Using as a query a single disease . . . 18
3.2.2 Plotting results of a Single Disease Query . . . 20
3.2.3 Using a list of diseases as a query . . . 21
3.2.4 Ploting results: Multiple Diseases . . . 22
3.2.5 Barplot according to number of publications that support the GDA . . . 24

4 Analyze the association between two diseases from the molecular perspective 26
4.1 Using the Jaccard Index . . . 26
4.2 Plotting results: Jaccard Index . . . 28

5 Warnings 30

1 Introduction

The psygenet2r package contains functions to query PsyGeNET [1], a resource on psychiatric diseases and their genes. The psygenet2r package includes analysis functions to study psychiatric diseases, their genes and disease comorbidities. A special focus is made on visualization of the results, providing a variety of representation formats such as networks, heatmaps and barplots.

1.1 Background

During the last years there has been a growing interest in the genetics of psychiatric disorders, leading to a concomitant increase in the number of publications that report these studies [2]. However, there is still limited understanding on the celular and molecular mechanisms leading to psychiatric diseases, which has limited the application of this wealth of data in the clinical practice. This situation also applies to psychiatric comorbidities. Some of the factors that explain the current situation is the heterogeneity of the information about psychiatric disorders and its fragmentation into knowledge silos, and the lack of resources that collect these wealth of data, integrate them, and supply the information in an intuitive, open access manner to the community. PsyGeNET has been developed to fill this gap. psygenet2r has been developed to facilitate statistical analysis of PsyGeNET data, allowing its integration with other packages available in R to develop data analysis workflows.

PsyGeNET is a resource for the exploratory analysis of psychiatric diseases and their associated genes. It is focused on eight psychiatric disorders: alcohol use disorders, bipolar disorders and related disorders, depressive disorders, schizophrenia spectrum and other psychotic disorders, cocaine use disorders, substance/drug induced depressive disorder, cannabis use disorders and drug-induced psychosis. PsyGeNET allows the exploration of the molecular basis of psychiatric disorders by providing a comprehensive set of genes associated to each disease. Moreover, it allows the analysis of the molecular mechanisms underlying psychiatric disease comorbidities. PsyGeNET database is the result of the data extracted from the literature by text mining using BeFree [4], followed by manual curation by domain experts. A team of 22 experts of different fields in psychiatry and neurosciences, defined each psychiatric disorder in terms of Concept Unique Identifiers (CUI) from the Unified Medical Language System (UMLS) and reviewed the association.The current version of PsyGeNET (v02) contains 3,771 associations, between 1,549 genes and 117 psychiatric disease concepts.

With psygenet2r package the user will be able to submit queries to PsyGeNET from R, perform a variety of analysis on the data, and visualize the results through different types of graphical representations.

The tasks that can be performed with psygenet2r package are the following:

1. Retrieve Gene-Disease Associations (GDAs) from PsyGeNET using as query a gene or a disease (single or a set of genes/diseases) of interest
2. Visualize the results according to the GDAs' attributes: PsyGeNET score, number of publications, sentences that report the GDA, source dadatabase
3. Visualize the results accoring to the disease (disease class) or gene (Panther class) attributes
4. Analyze the association between two diseases from the molecular perspective (using the Jaccard index).

In the following sections the specific functions that can be used to address each of these tasks are presented.

1.2 Installation

The package psygenet2r is provided through Bioconductor. To install psygenet2r the user must type the two following commands in an R session:

```
source( "http://bioconductor.org/biocLite.R" )
biocLite( "psygenet2r" )
```

```
library( psygenet2r )
```

1.3 DataGeNET.Psy object

DataGeNET.Psy object is obtained when psygenetGene and psygenetDisease functions are applied. This object is used as input for the rest of psyGeNET2r functions, like the plot function.

DataGeNET.Psy object contains all the information about the different diseases/genes associated with the gene/disease of interested retrieved from PsyGeNET. It object contains a summary of the search, such as the search input (gene or disease), the selected database, the gene or disease identifier, the number of associations found (N. Results) and the number of unique results obtained (U. Results).

```
t1

## Object of class 'DataGeNET.Psy'
##   . Type:         gene
##   . Database:     ALL
##   . Term:         4852
##   . N. Results:   13
##   . U. Diseases:  13
##   . U. Genes:     1

class(t1)

## [1] "DataGeNET.Psy"
## attr(,"package")
## [1] "psygenet2r"
```

This object comes with a series of function to allows users to interact with the information retireved from PsyGeNET. These functions are ngene, ndisease, extract and plot. The first function ngene return the number of retrieved genes for a given query. ndisease is the homologous function but for the diseases. The function extract returns a formatted data.frame with the complete set of information downloaded from PsyGeNET. Finally, the plot function lets visualize the results in a variety of ways such as gene-disease association networks or heatmaps.

2 PsyGeNET and psygenet2r

The PsyGeNET web interface can be explored by searching a specific gene or a specific disease, and psygenet2r package has the same options. Therefore, the starting point for psygenet2r are psygenetGene and psygenetDisease functions.

PsyGeNET data is classified according to the database used as a source of information ("source database"). Therefore, any query run on PsyGeNET requires to specify the source database using the argument called database. Table 1 shows the source databases in PsyGeNET and their description. By default, the database ALL is used in psygenet2r. For illustrating purposes along the vignette, database ALL will be used in most of code snippets.

Table 1: Source databases included in PsyGeNET

Name	Description
psycur15	Psychiatric disorders Gene association manually curated first release (2 experts)
psycur16	Psychiatric disorders Gene association manually curated second release (22 experts)
ALL	All previous Databases

3 Retrieve gene-disease associations (GDAs) from psygenet2r

3.1 Using genes as a query

psygenet2r package allows exploring PsyGeNET information using a specifc gene or a list of genes. It retrieves the information that is available in PsyGeNET (associated diseases, source database, PsyGeNET score, number of publications, attributes of genes, etc) and allows to visualize the results in different ways.

3.1.1 Using as a query a single gene

In order to look for a single gene into PsyGeNET, we can use the psygenetGene function. This function retrieves PsyGeNET's information using both, the NCBI gene identifier and the official Gene Symbol from HUGO. It contains also other arguments like the database to query, the PsyGeNET score and the check argument, that allows the user to decide if he wants to validate or not the gene before interrogating the database.

As an example, the gene *NPY*, whose entrez id is *4852* is queried using psygenetGene function, and using alternatively the official HUGO Gene Symbol. In this example database "ALL".

```
t1 <- psygenetGene( gene = 4852,
                    database = "ALL",
                    check = FALSE)
t1

## Object of class 'DataGeNET.Psy'
##  . Type:          gene
##  . Database:      ALL
##  . Term:          4852
##  . N. Results:    13
##  . U. Diseases:   13
##  . U. Genes:      1
```

```
t2 <- psygenetGene( gene = "NPY", database = "ALL", check = FALSE )
t2

## Object of class 'DataGeNET.Psy'
##  . Type:          gene
##  . Database:      ALL
##  . Term:          NPY
##  . N. Results:    13
##  . U. Diseases:   13
##  . U. Genes:      1
```

Both cases result in an DataGeNET.Psy object:

```
class(t1)

## [1] "DataGeNET.Psy"
## attr(,"package")
## [1] "psygenet2r"

class(t2)

## [1] "DataGeNET.Psy"
## attr(,"package")
## [1] "psygenet2r"
```

In the particular example used, by inspecting the `DataGeNET.Psy` object, we can see that the gene *NPY* is associated to 13 different diseases in PsyGeNET (with no restriction on the PsyGeNET score).

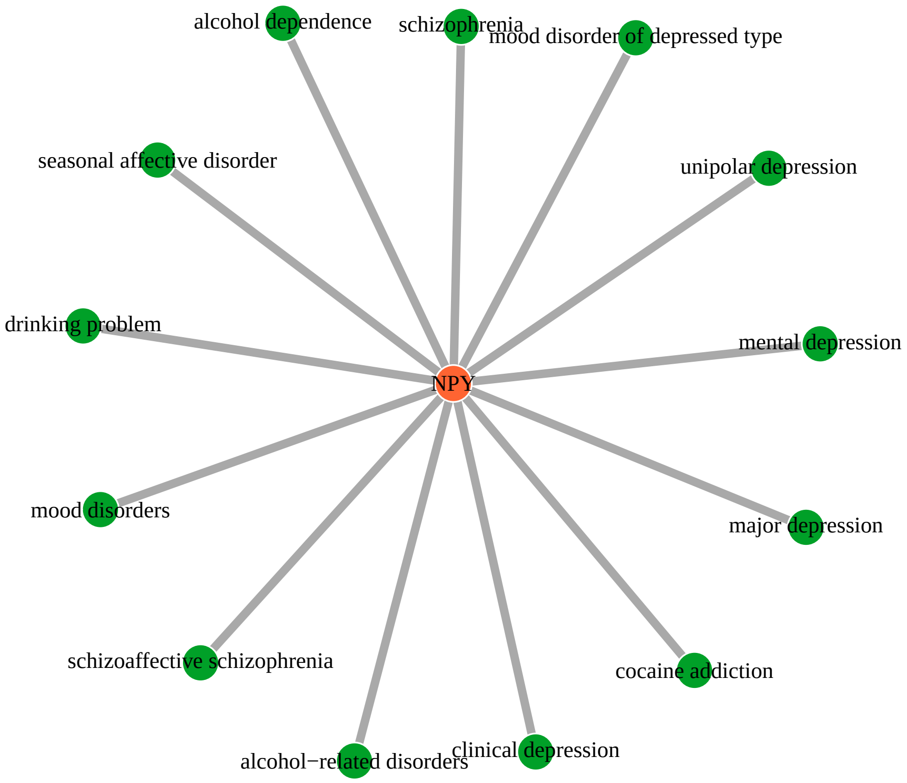
3.1.2 Ploting the results of a Single Gene Query

`psygenet2r` offers several options to visualize the results from PysGeNET: a network showing the diseases related to the gene of interest, or a network showing the strength of the association between the three main psychiatric disorders in PsyGeNET and the gene of interest. Each one of these graphics can be obtained changing the type argument.

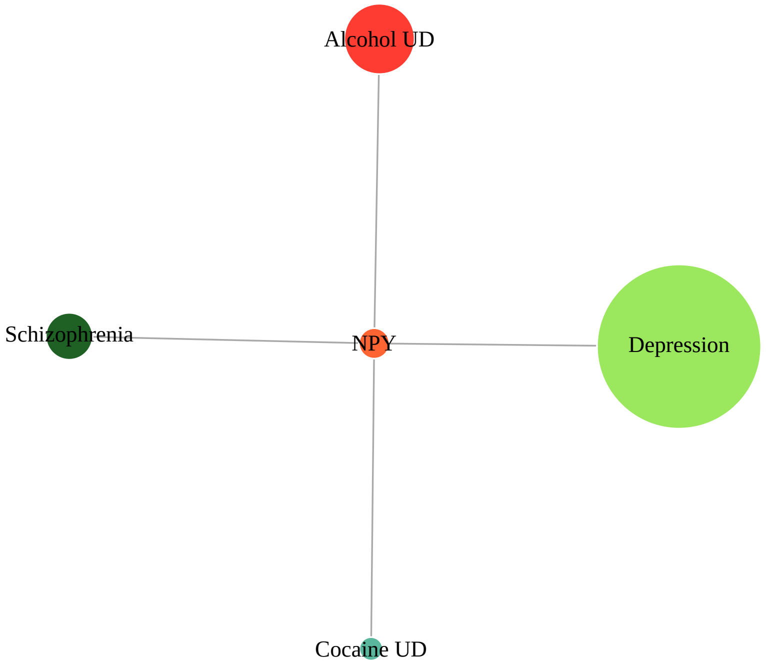
By default, **`psygenet2r`** shows this type of network when ploting a `DataGeNET.Psy` object obtained by a gene-query. The result is a network where green nodes are diseases and the orange node is the gene of interest.

On the other hand, results can be visualized according to the 3 psychiatric disorders classes available in PsyGeNET (major depression, alcohol use disorder and cocaine use disorder) setting the `type` argument to `"disease class"`. As a result, a network with 4 nodes is obtained. The node's size of each psychiatric disorder is proportional to the number of disease concepts that belongs to each disease class, from the total number of diseases associated to the gene.

```
plot( t1, type = "individual disease" )
```



```
plot( t1, type = "disease class" )
```



In our example, NPY is associated to the three psychiatric disorders, with an important contribution of major depression disorders.

3.1.3 Using as a query a list of genes

In the same way, **psygenet2r** allows to query PsyGeNET given a list of genes of interest. The same function, **psygenetGene**, accepts a vector of NCBI gene identifiers or HUGO official gene symbols.

To illustrate this functionality, a list of 20 genes was extracted from the article entitled *"The Genetics of Major Depression"* [5], where these genes are associated to depression. The vector of genes can be defined as follows:

```
genesOfInterest <- c( "COMT", "CLOCK", "DRD3", "GNB3", "HTR1A",
                      "MAOA", "HTR2A","HTR2C", "HTR6", "SLC6A4",
                      "ACE",  "BDNF", "DRD4", "HTR1B", "HTR2B",
                      "HTR2C", "MTHFR", "SLC6A3", "TPH1", "SLC6A2",
                      "GABRA3"
)
```

Then, the function `psygenetGene` is applied. In this case an extra argument called `verbose` was set to `TRUE`. This shows some information during the query-process, for example the message informing that there are repeated genes in the list (the gene *HTR2C* was placed twice in the list to raise this message).

```
m1 <- psygenetGene(
    gene     = genesOfInterest,
    database = "ALL",
    verbose  = TRUE,
    check = FALSE
)

## Warning in psygenetGene(gene = genesOfInterest, database = "ALL", verbose = TRUE,
:  Removing duplicates from input genes list.
## Staring querying PsyGeNET for COMT, CLOCK, DRD3, GNB3, HTR1A, MAOA, HTR2A, HTR2C,
HTR6, SLC6A4, ACE, BDNF, DRD4, HTR1B, HTR2B, MTHFR, SLC6A3, TPH1, SLC6A2, GABRA3 in
ALL database.
## Warning in psygenetGene(gene = genesOfInterest, database = "ALL", verbose = TRUE,
:  One or more of the given genes is not in PsyGeNET ( 'ALL' ):
##    - HTR2B
```

A `DataGeNET.Psy` object is obtained. In this particular example, 20 genes are present in PsyGeNET and are associated to 34 diseases, involving 238 GDAs.

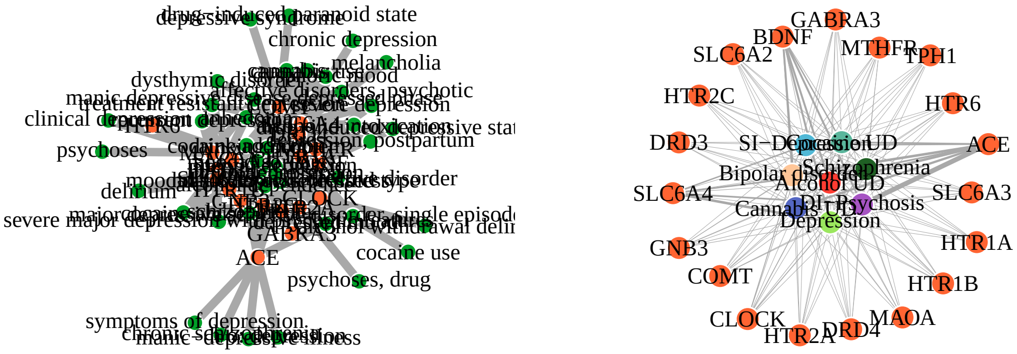
```
m1

## Object of class 'DataGeNET.Psy'
##   . Type:         gene
##   . Database:     ALL
##   . Term:         COMT ... GABRA3
##   . N. Results:   212
##   . U. Diseases:  42
##   . U. Genes:     19
```

3.1.4 Ploting the results of the query using a list of genes

`psygenet2r` provides several options to visualize the results of these queries, such as networks, heatmaps and venn diagrams.

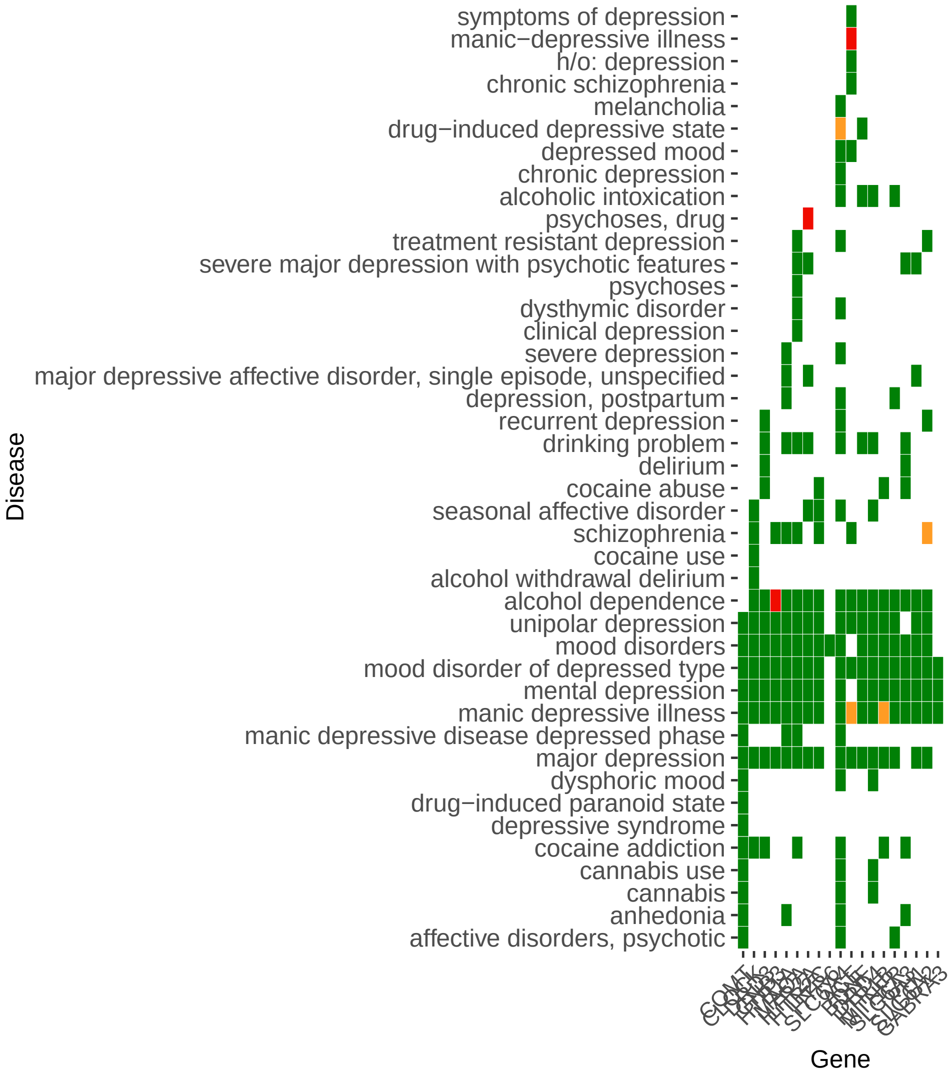
As for single gene, the default option in `psygenet2r` results in a network chart, where the green nodes belong to diseases and the oranges nodes belong to genes. It is also possible to visualize the results by grouping the diseases according to the psychiatric disorders present in PsyGeNET specifying the argument type.

```r
layout( matrix( c( 1, 2 ), nrow = 1 ) )
plot( m1 )
plot( m1, type = "disease class" )
```



psygenet2r package allows to visualize the GDAs attributes in a heatmap. The argument type must be "heatmapGenes" and the PsyGeNET score can also be determined by the user setting the cut-off argument to the score of interest. In this example, the cut-off is set to 0.5, it means that only those associations with an score equal or higher to 0.5 will be shown. **Warning**: if there are more than 100 associations, only the first 100 with highest score are shown in the heatmap. In this kind of representation we can identify genes that are associated to several diseases (e.g. SLC6A4) and also genes that are specific for a particular diseases (e.g. GABRA3).

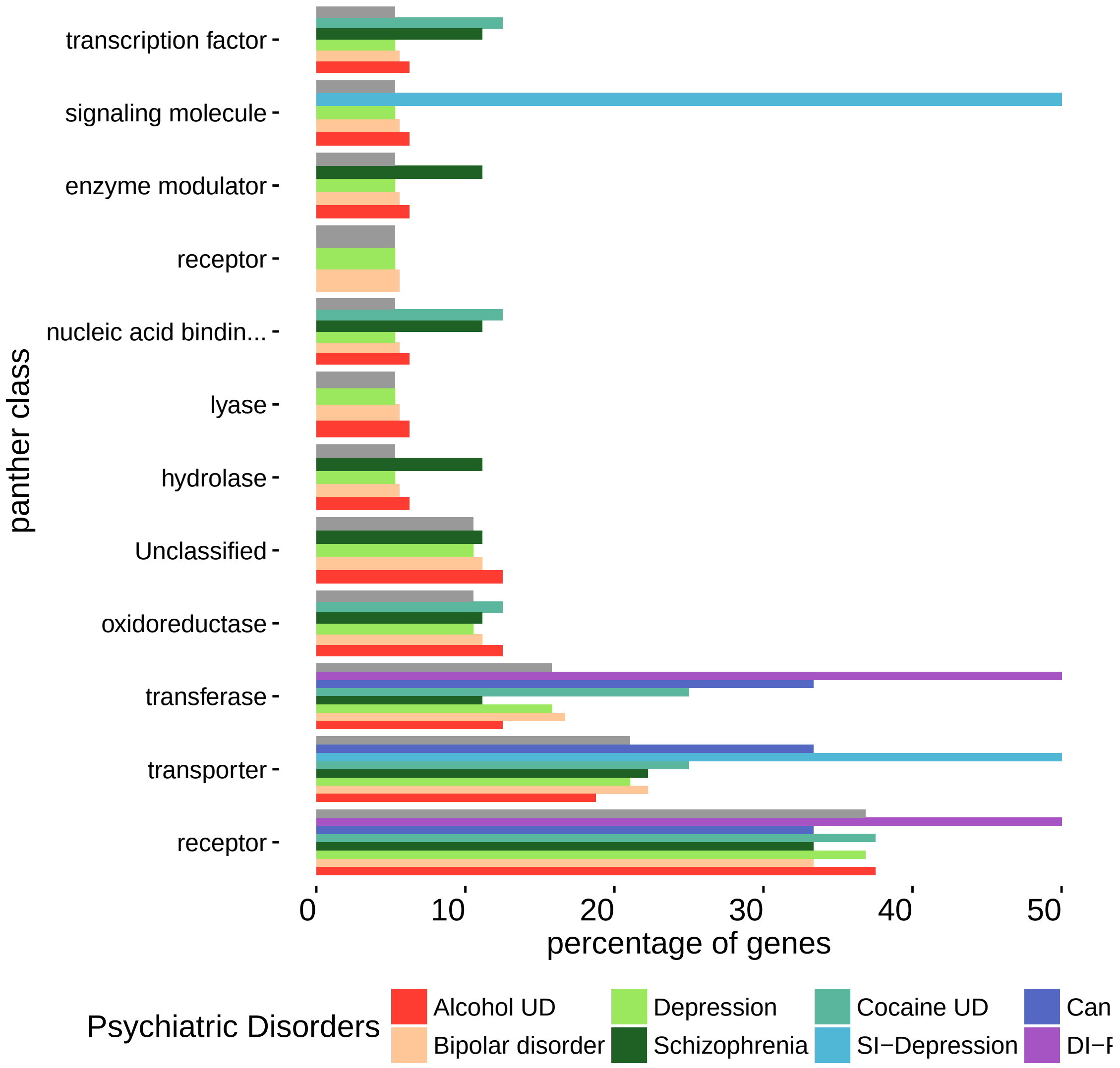
```r
plot( m1, type="heatmapGenes" )
```



psygenet2r package also allows to analyze a gene list according to the function of the proteins encoded by these genes. The PANTHER Protein Class Ontology classifies proteins according to their function.

The pantherGraphic function shows the Panther class to which the proteins belong according to their associated psychiatric disorder. It provides a graphic with the results of these analysis, being the input a list of genes and the database (ALL, CURATED, PsyCUR, etc). The input genes can be from a vector that contains the genes of interest, or from the genes obtained in the DataGeNET.psy object in a disease or disease-list query. An score argument can be added to filter results. It can also be done given a DataGeNET.psy object obtained by querying with a single gene.

```
genesOfInterest <- unique( genesOfInterest )
pantherGraphic( genesOfInterest, "ALL", check = FALSE )

## Warning in psygenetGene(geneList, database, verbose = verbose, hostMart = hostMart,
:  One or more of the given genes is not in PsyGeNET ( 'ALL' ):
##    - HTR2B
```

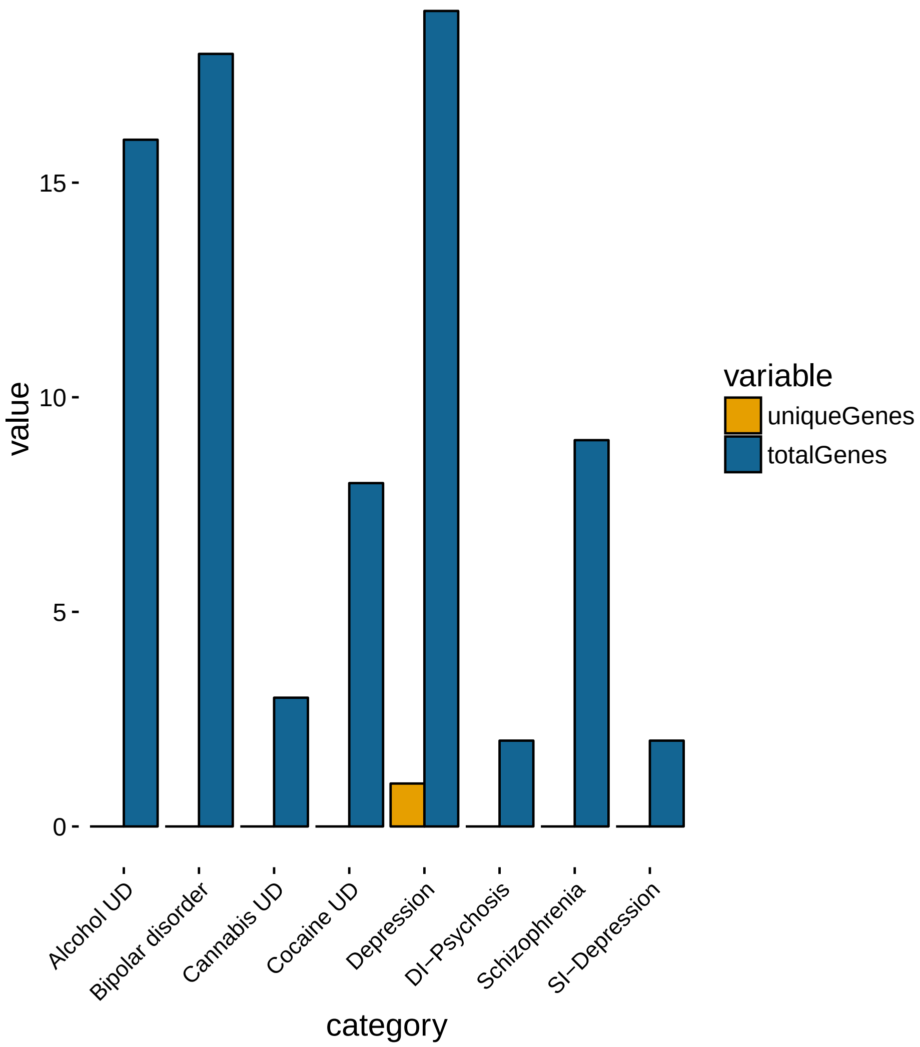


3.1.5 Ploting results of the Multiple Gene Query

In addition, the results of a query using a list of genes can be visualized using barplots and pie charts.

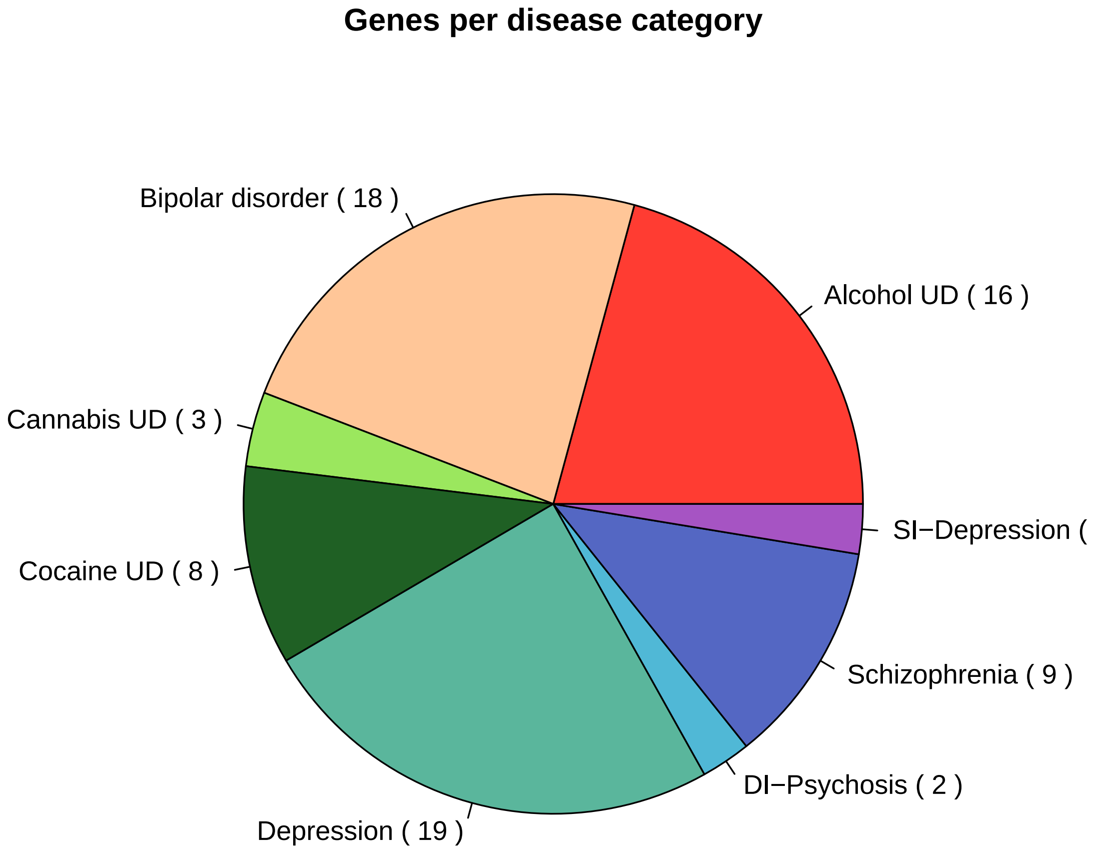
Here, in this first barplot we show how many of the genes that are associated with a disease category are exclusively associated to it and how many of them are also associated with any other category.

```
geneAttrPlot( m1, type = "category" )
```



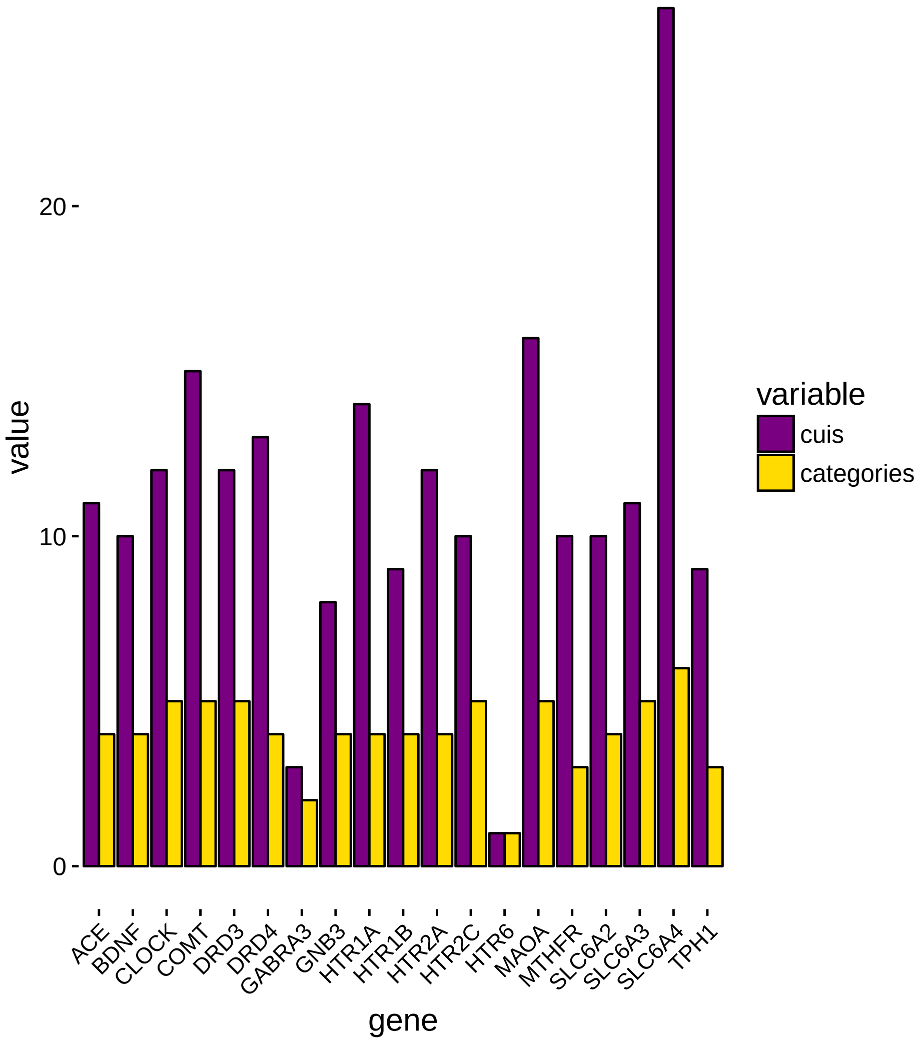
If we are only interested in how many of our input genes are associated to each disease category, a pie chart can be obtained applying the `plot` function and setting up the `type` argument to `pieChart`.

```
geneAttrPlot( m1, type = "pie" )
```



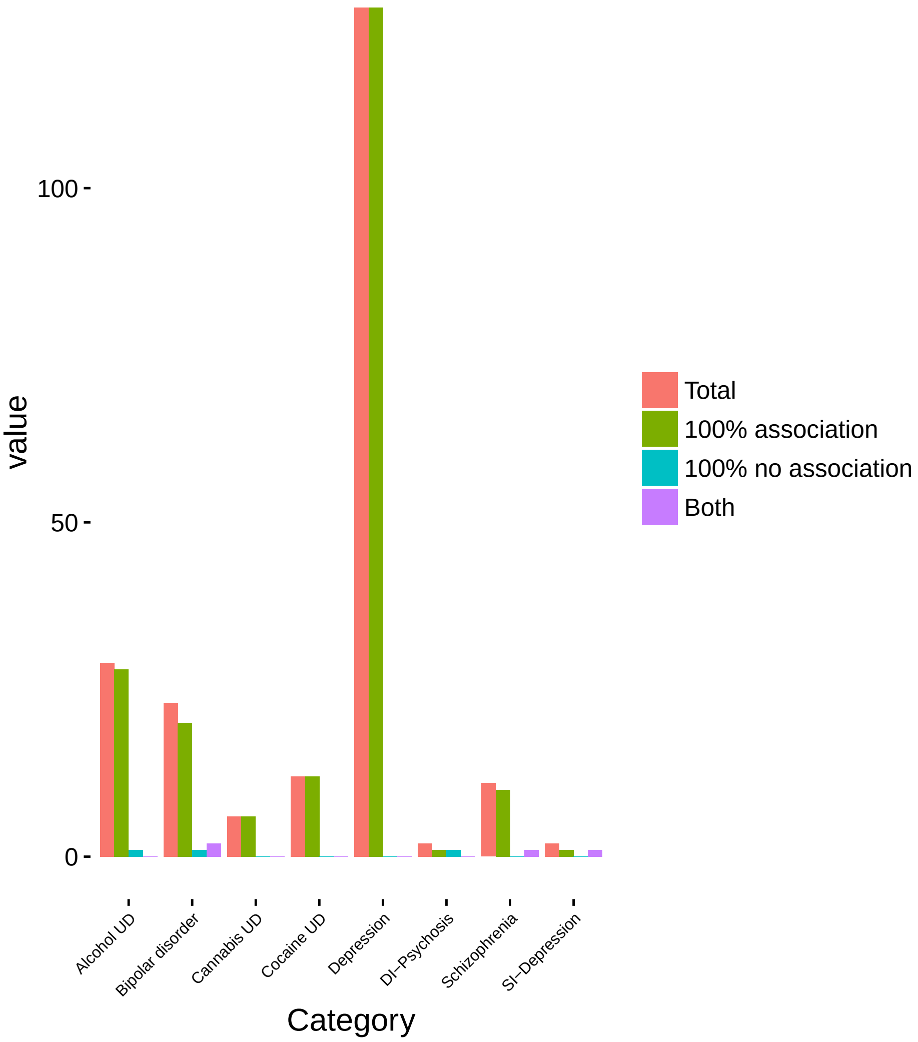
On the other hand, this second barplot show us the number of cuis and disease categories associated to each one of our genes of interest.

```
geneAttrPlot( m1, type = "gene" )
```



An alternative graphical representation is a barplot showing gene-disease association according to each disease category taking into account the type of evidence of each assocation (association, no association, both).

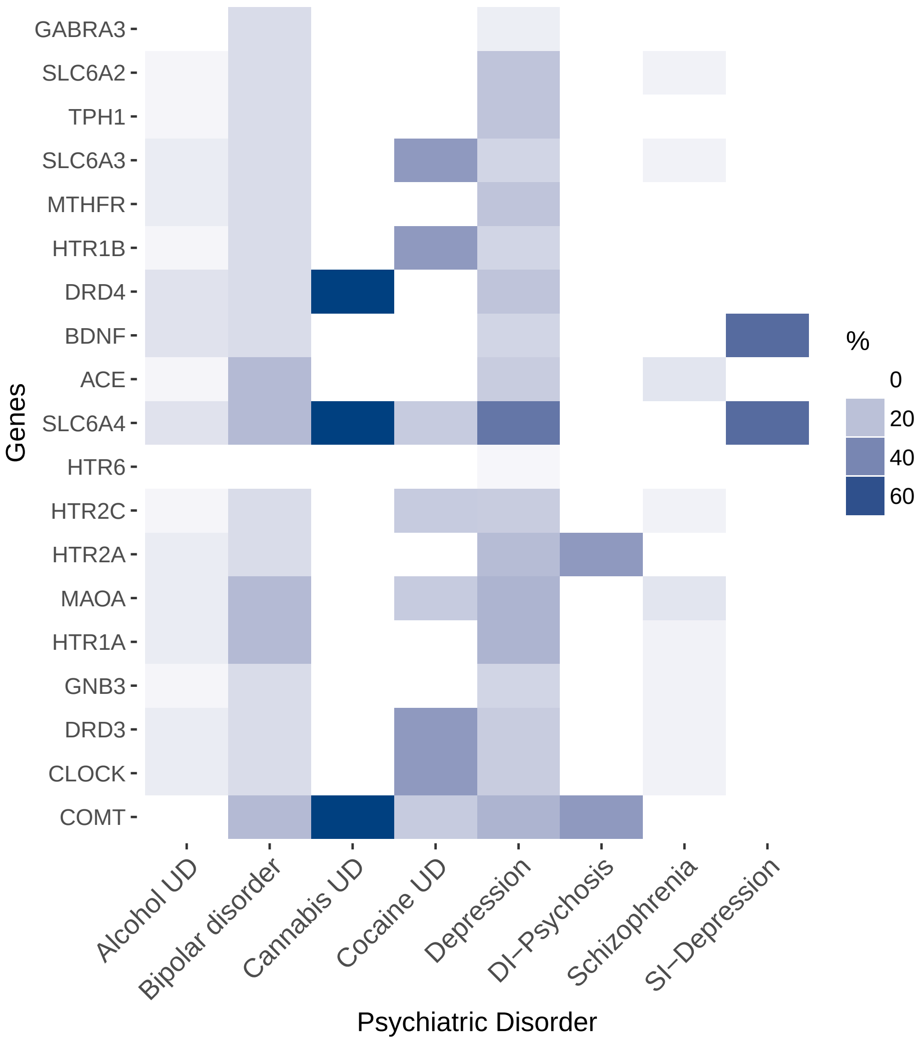
```
geneAttrPlot( m1, type = "index" )
```



The other one allows to analyze if the genes that are being studied present a specific association with a subtype of disorder or if they are associated with several of them in the same psychiatric disorder. The percentage of diseases to which a gene appear associated with each psychiatric disorder is estimated. This percentage is relative to the total possible subtypes of disorders present in PsyGeNET (3 for cocaine use disorder, 7 for alcoholism use disorder and 27 for major depression). The resultant values are represented in a heatmap according to a blue color scale.

In order to obtain this graphic, `type` must be set to `"heatmap"`. The resulting heatmap shows which are the genes that are higher or lower associated to each one of the psychiatric disorders.

```
plot( m1, type = "heatmap" )

## Warning:  Non Lab interpolation is deprecated
```



As it could be expected for the genes that are used as genes of interest, all of them are associated to major depression, to a greater or lesser extent. Some of them are also associated to the other two psychiatric disorders.

3.1.6 Enrichment analysis for a list of genes

The R package `psygenet2r` allows to perform an enrichment analysis on a list of genes with PsyGeNET diseases. It is done by using the function `enrichedPD`. In order to illustrate this function, a set of 8 genes extracted from a GWAS study [6] was selected.

```
genesOfInterest <- c("PECR", "ADH1C", "CAST", "ERAP1", "PPP2R2B",
                     "ESR1", "GATA4", "CDH13")
tbl <- enrichedPD( genesOfInterest, database = "ALL")
tbl

##                                                  MPD    p.value
## 1                              Alcohol use disorders 0.22379158
## 2               Bipolar disorders and related disorders 0.80106155
## 3                               Depressive disorders 0.47914172
## 4 Schizophrenia spectrum and other psychotic disorders 0.80090723
## 5                                Cocaine use disorders 1.00000000
## 6                Substance-induced depressive disorder 1.00000000
## 7                               Cannabis use disorders 0.06553039
## 8                          Substance Induced-Psychosis 1.00000000
```

```
genesOfInterest <- c(55825, 126, 831, 51752, 5521)
tb2 <- enrichedPD( genesOfInterest, database = "ALL")
tb2

##                                                    MPD   p.value
## 1                               Alcohol use disorders 0.4029606
## 2              Bipolar disorders and related disorders 1.0000000
## 3                                Depressive disorders 1.0000000
## 4 Schizophrenia spectrum and other psychotic disorders 0.8262320
## 5                                Cocaine use disorders 1.0000000
## 6                Substance-induced depressive disorder 1.0000000
## 7                               Cannabis use disorders 1.0000000
## 8                          Substance Induced-Psychosis 1.0000000
```

The result is a table with a p-value of the enrichment of the given list of genes for each psychiatric disorder in PsyGeNET. As we can see these genes are enriched in the alcohol use disorder genes of PsyGeNET (p-val = 0.0009). This result is in agreement to what Treutlein et al. find in their analysis, in which they determined that these genes where associated with alcohol dependence.

3.1.7 Enrichment analysis based on anatomical terms (TopAnat) for a list of genes

```
genesOfInterest <- c( "COMT", "CLOCK", "DRD3", "GNB3", "HTR1A",
                      "MAOA", "HTR2A","HTR2C", "HTR6", "SLC6A4",
                      "ACE",  "BDNF", "DRD4", "HTR1B", "HTR2B",
                      "HTR2C", "MTHFR", "SLC6A3", "TPH1", "SLC6A2",
                      "GABRA3"
)
tpAnat <- topAnatEnrichment( genesOfInterest, cutOff = 1 )

## Error in stri_replace_all_regex(x, c("\\$", "\\\\(\\d)"), c("\\\\$", "\\$$1"), :
object 'database' not found

head( tpAnat )

## Error in head(tpAnat):  object 'tpAnat' not found
```

3.1.8 Sentences that report a GDA

psygenet2r package also allows extract the pmids sentences that report a gene-disease asssociation. It is done by using two different functions, psygenetGeneSentences and extractSentences. psygenetGeneSentences needs as input a gene list and a database to query in. The output of this function is a DataGeNET.Psy object. This object is passed to the extractSentences function, that also needs the disorder of interest.

```
genesOfInterest
```

```
##  [1] "COMT"   "CLOCK"  "DRD3"   "GNB3"   "HTR1A"  "MAOA"   "HTR2A"
##  [8] "HTR2C"  "HTR6"   "SLC6A4" "ACE"    "BDNF"   "DRD4"   "HTR1B"
## [15] "HTR2B"  "HTR2C"  "MTHFR"  "SLC6A3" "TPH1"   "SLC6A2" "GABRA3"

sss <- psygenetGeneSentences( geneList = genesOfInterest,
                              database = "ALL")

## Warning in psygenetGeneSentences(geneList = genesOfInterest, database = "ALL"):
One or more of the given genes is not in PsyGeNET ( 'ALL' ). Genes:  HTR2B

sss

## Object of class 'DataGeNET.Psy'
##  . Type:         gene
##  . Database:     ALL
##  . Term:         COMT ... SLC6A2
##  . N. Results:   778
##  . U. Diseases:  42
##  . U. Genes:     18

geneSentences <- extractSentences( object = sss,
                                   disorder = "alcohol dependence")
dim(geneSentences)

## [1] 75  8
```

The result is a data frame that contains the gene, disease, original db, the pmid and the sentence.

3.2 Using diseases as a query

psygenet2r package allows to explore PsyGeNET information searching a specifc disease or a list of diseases. As in the case of genes, it retrieves the information that is available in PsyGeNET and allows to visualize the results in several ways.

3.2.1 Using as a query a single disease

In order to look for a single disease into PsyGeNET, psygenet2r has the psygenetDisease function. This function allows you to obtain PsyGeNET's information using both disease id or disease name, and the database as input (by default is CURATED). **Warning**: if user uses the disease name, it must be written in the same way that appears in PsyGeNET, otherwise the disorder will not be found (Alba: i think this warning can be removed, since you provide a function to search for the standard name and chi of a disease).

If the user do not know the disease identifier, it can use the getUMLS function to obtain disease names and UMLS CUIs from a string query. Providing as input the term and source of interest, getUMLs function retrieves all the PsyGeNET concepts that contain it. As an example it is shown the query results for depressive term in ALL databases.

```
getUMLs("depressive", database="ALL")

##                                                             DiseaseName
## 2                                              manic depressive illness
## 6      major depressive affective disorder, single episode, unspecified
## 103                               manic depressive disease depressed phase
## 247  major depressive affective disorder, recurrent episode, unspecified
## 288                                             manic-depressive illness
## 462                                        drug-induced depressive state
## 1698                                                 depressive syndrome
## 2048                          recurrent depressive disorder, unspecified
##                          PsychiatricDisorder          umls
## 2    Bipolar disorders and related disorders uml s:C0005586
## 6                       Depressive disorders umls:C0024517
## 103  Bipolar disorders and related disorders umls:C0005587
## 247                      Depressive disorders umls:C0154409
## 288  Bipolar disorders and related disorders umls:C1839839
## 462    Substance-induced depressive disorder umls:C0338715
## 1698                     Depressive disorders umls:C0086133
## 2048                     Depressive disorders umls:C0349218
```

As an example, the disease *Single Major Depressive Episode*, whose disease id is *umls:C0024517* is queried using `psygenetDisease` function, and using both, disease name and disease id. For this example database `"ALL"` is selected:

```
d1 <- psygenetDisease( disease  = "umls:C0024517",
                       database = "ALL",
                       score    = c('>', 0.5 ) )
d1

## Object of class 'DataGeNET.Psy'
##   . Type:          disease
##   . Database:      ALL
##   . Term:          umls:C0024517
##   . N. Results:    9
##   . U. Genes:      9
##   . U. Diseases:   1
```

```
d2 <- psygenetDisease( disease = "major depressive affective disorder, single episode, unspecifie
                       database = "ALL",
                       score    = c('>', 0 ) )
d2

## Object of class 'DataGeNET.Psy'
##   . Type:          disease
##   . Database:      ALL
##   . Term:          major depressive affective disorder, single episode, unspecified
##   . N. Results:    9
##   . U. Genes:      9
##   . U. Diseases:   1
```

Both cases result in an `DataGeNET.Psy` object, that contains the same information as in the gene query search:

```
class(d1)

## [1] "DataGeNET.Psy"
## attr(,"package")
## [1] "psygenet2r"

class(d2)
```

```
## [1] "DataGeNET.Psy"
## attr(,"package")
## [1] "psygenet2r"
```

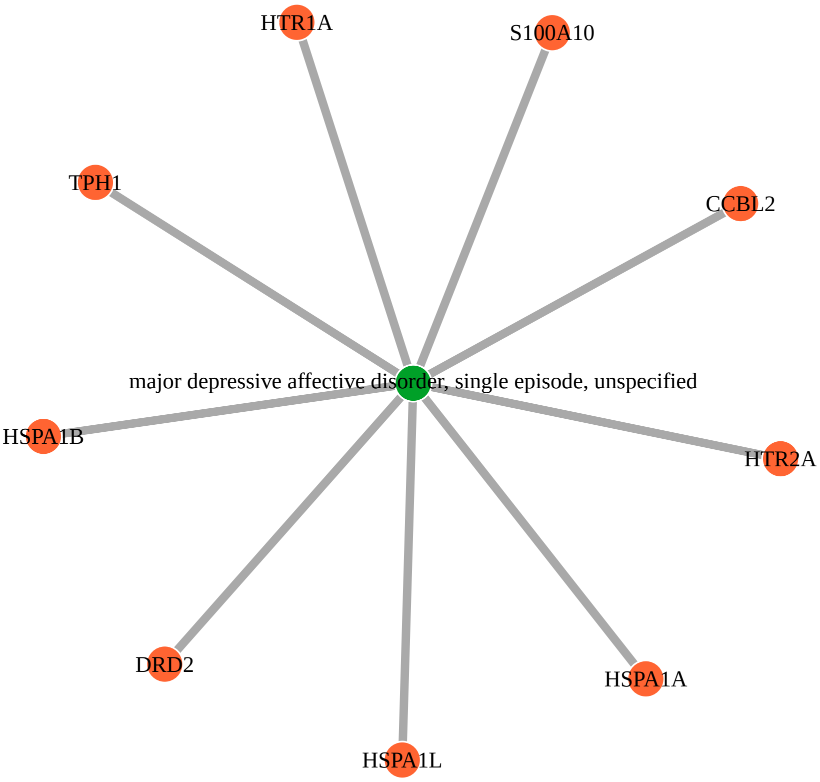
The argument `score` is filled with a vector which first position can be '<' or '>' to indicate if the threshold is read as lower or upper. The second argument is the threshold in itself which will always be included. This argument is also present in `psygenetGene`.

3.2.2 Plotting results of a Single Disease Query

`psygenet2r` package offers several options to visualize the results from PysGeNET given a disease: a network showing the genes related to the disease of interest and a barplot showing how many publications report each one of the gene-disease associations.

By default, `psygenet2r` shows the GDAs network when ploting a `DataGeNET.Psy` object with a disease-query. The result is a network where, orange nodes are genes and the central and green node is the disease of interest.

```
plot ( d1 )
```


HTR1A
S100A10
TPH1
CCBL2
major depressive affective disorder, single episode, unspecified
HSPA1B
HTR2A
DRD2
HSPA1A
HSPA1L

3.2.3 Using a list of diseases as a query

In the same way, psygenet2r allows to query PsyGeNET given a set of diseases of interest. The same function, psygenetDisease, accepts a vector of disease-names or disease-ids (umls code).

To illustrate this functionality, two disorders has been selected: bipolar disorder and major depressive disorder. The vector of diseases can be defined for example, as follows:

```
diseasesOfInterest <- c( "manic depressive illness","major depression" )
```

```
tt <- psygenetDisease( disease  = diseasesOfInterest,
                       database = "ALL" )
tt

## Object of class 'DataGeNET.Psy'
##   . Type:         disease
##   . Database:     ALL
##   . Term:         manic depressive illness ... major depression
##   . N. Results:   756
##   . U. Genes:     639
##   . U. Diseases:  2
```

```
dm <- psygenetDisease( disease  = c( "umls:C0005586", "umls:C1269683" ),
                       database = "ALL" )
dm

## Object of class 'DataGeNET.Psy'
##   . Type:         disease
##   . Database:     ALL
##   . Term:         umls:C0005586 ... umls:C1269683
##   . N. Results:   756
##   . U. Genes:     639
##   . U. Diseases:  2
```

```
tm <- psygenetDisease( disease  = c( "manic depressive illness","umls:C1269683" ),
                       database = "ALL" )
tm

## Object of class 'DataGeNET.Psy'
##   . Type:         disease
##   . Database:     ALL
##   . Term:         manic depressive illness ... umls:C1269683
##   . N. Results:   756
##   . U. Genes:     639
##   . U. Diseases:  2
```

Three cases result in an DataGeNET.Psy object:

```
class(tt)

## [1] "DataGeNET.Psy"
## attr(,"package")
## [1] "psygenet2r"
```

```
class(dm)

## [1] "DataGeNET.Psy"
## attr(,"package")
## [1] "psygenet2r"

class(tm)

## [1] "DataGeNET.Psy"
## attr(,"package")
## [1] "psygenet2r"
```

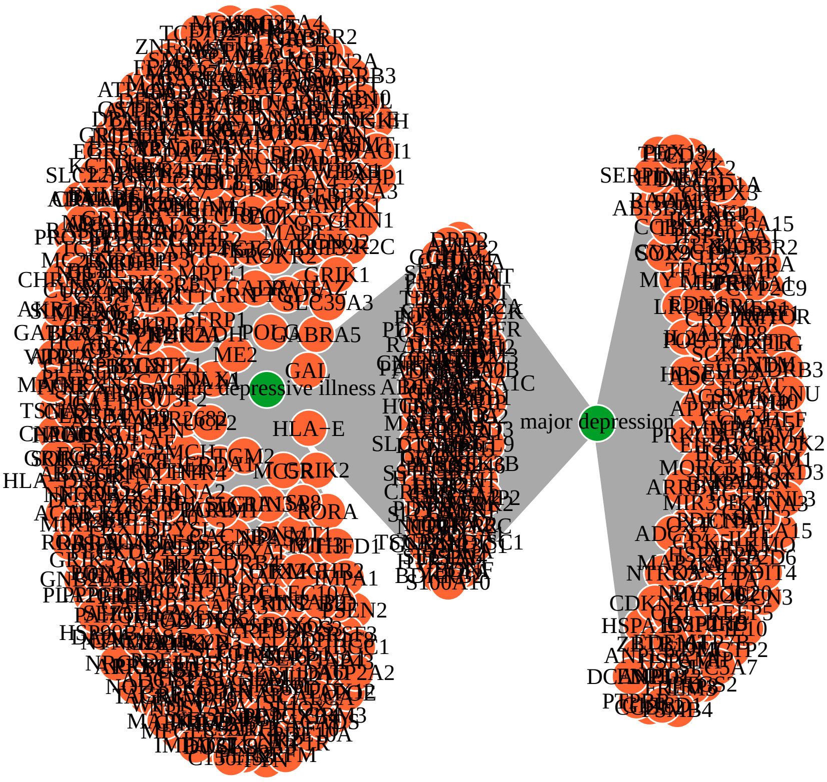
This type of object contains all the information about the different genes associated with the diseases of interest retrieved from PsyGeNET. By inspecting the `DataGeNET.Psy` object we can see that, according to PsyGeNET and querying in ALL databases, the 2 disorders of interest are associated to 494 different genes in 700 different associations.

3.2.4 Ploting results: Multiple Diseases

`psygenet2r` provides a network graphic and a heatmap to visualize the results of search with multiple input items.

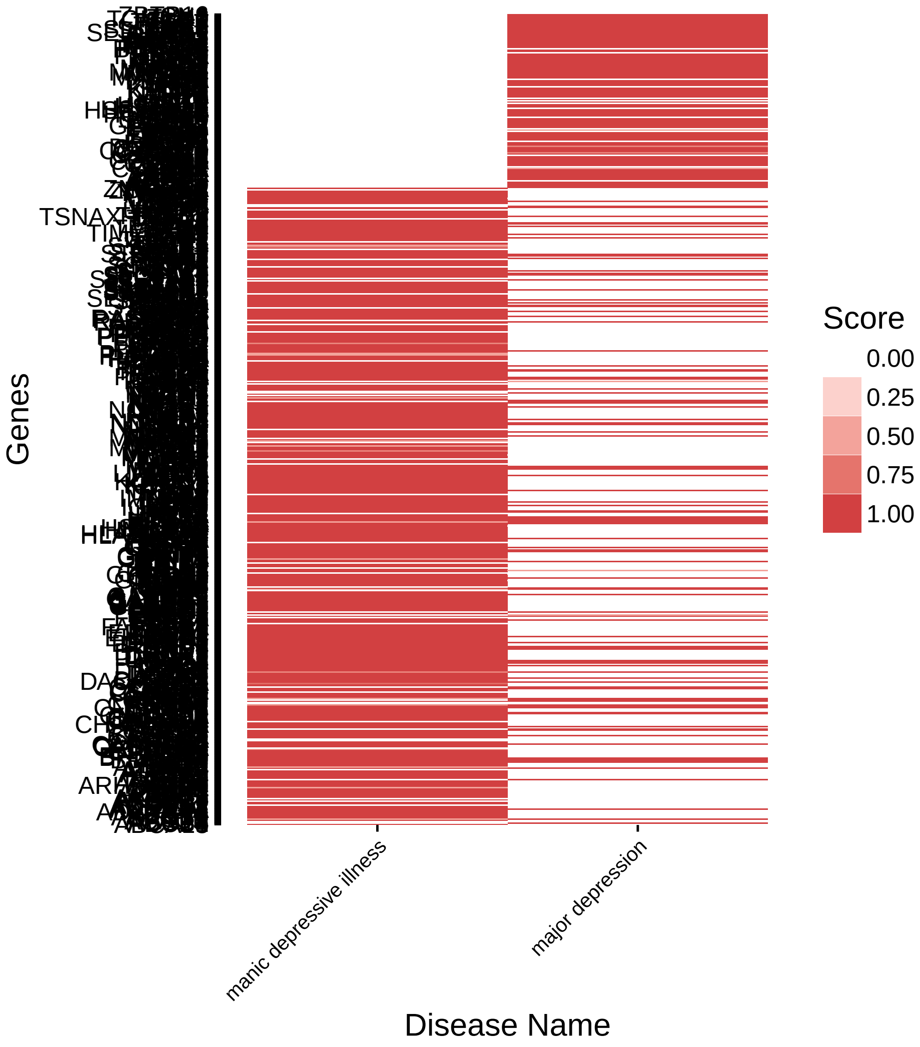
As for single disease, GDAs network is the default option in `psygenet2r`. In the resulting network chart, the green nodes represent diseases and the oranges nodes represent genes.

```
plot( tm )
```



Another possible option is visualize it in a heatmap. The argument `type` can be set to `"heatmap"` and the score argument must be determined by the user. For example, in the previous search, 494 different genes were found associated to the disorders of interest. The PsyGeNET score ranges from 0 to 1; greater score means that there are more evidences supporting the association, so we can restrict our heatmap to those association with an score greater or equal to 0.7. The darkest red belongs to the greatest score.

```
plot( tm, type = "heatmap" )

## Warning:  Non Lab interpolation is deprecated
```

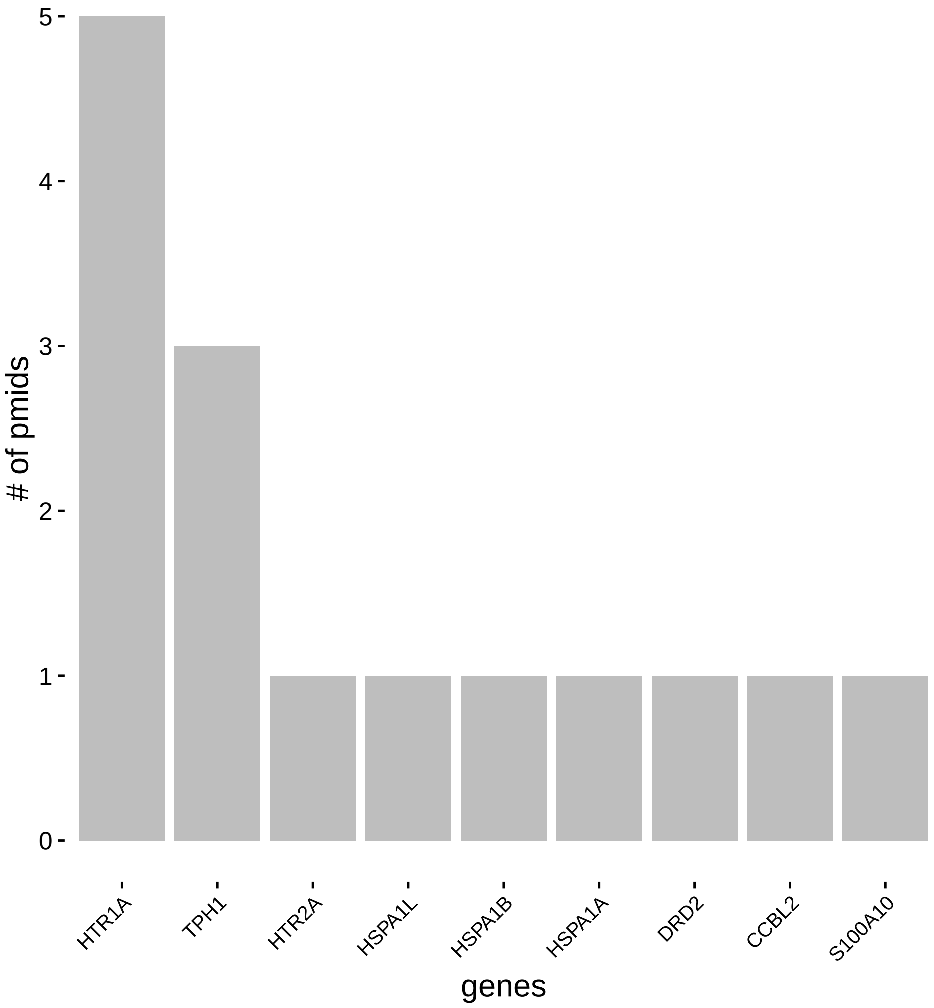


The result is a heatmap where the genes are located at X axis, and disordes appear at Y axis. The red rank color is related to the PsyGeNET score, being the darkest one the association with the highest score.

3.2.5 Barplot according to number of publications that support the GDA

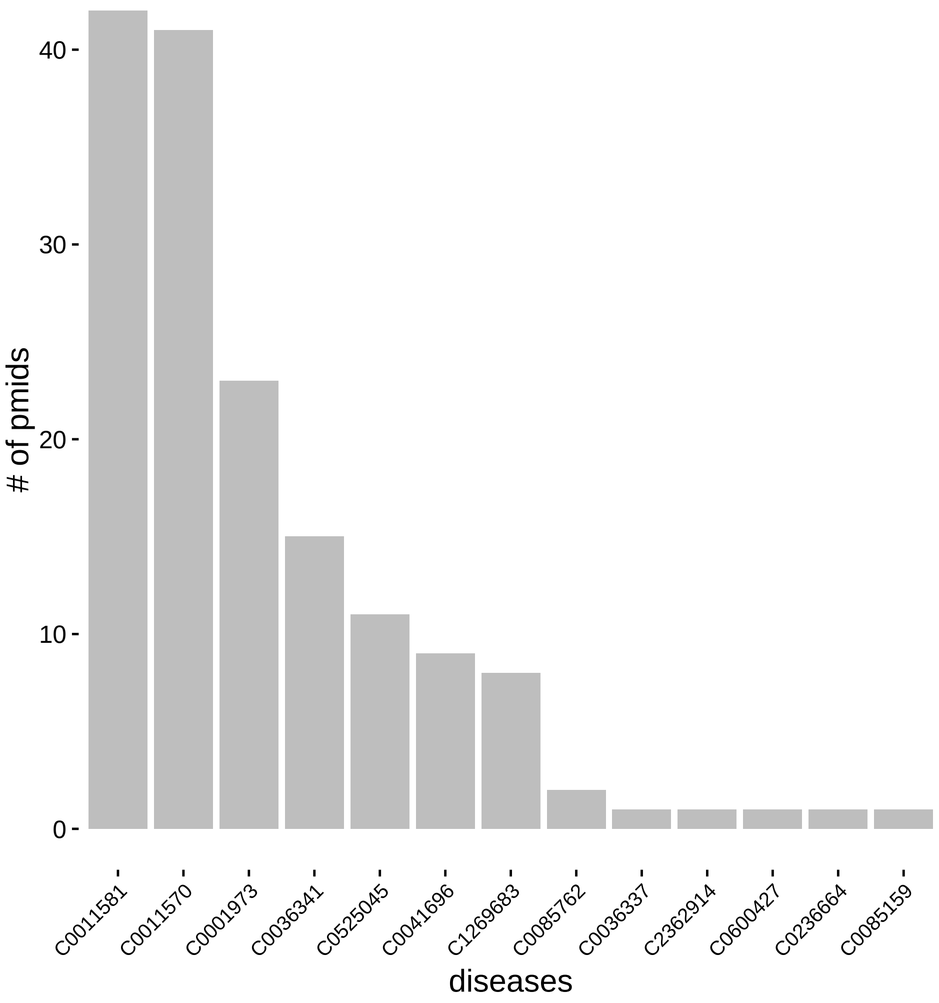
`psygenet2r` package allows to see how many pubmed ids support each gene-disease association. This can be visualize in a barplot by determining the gene or disease id in the `name` argument and setting `type` argument to `"barplot"`.

```
plot ( d1, name = "major depressive affective disorder, single episode, unspecified", type = "barp
```



As a result, a barplot is obtained. The X axis contains the genes related to the disease of interest, sorted by the number of pubmed ids in which we can find the gene-disease association. alternatively, the results can be visualized for the diseases.

```
plot ( t1, name = "NPY", type = "barplot" )
```



4 Analyze the association between two diseases from the molecular perspective

We can study the association between two diseases from the point of view of shared genetic contribution. More precisely, we can estimate the degree of association of two diseases by means of the number of genes that are shared by the two diseases, over the total number of disease genes. Similarity measures such as the Jaccard Index can be used to estimate disease similarity. The significance of the Jaccard index is estimated by a bootstrap procedure (see below).

4.1 Using the Jaccard Index

The Jaccard Index, also known as the Jaccard similarity coefficient, is a statistic used for comparing the similarity of two sets, and is defined as the size of the intersection divided by the size of the union of the sample sets:

$$J(A,B) = \frac{\mid A \cap B \mid}{\mid A \cup B \mid}$$

psygenet2r comes with functions to compute the Jaccard Index as an estimation of the similarity of two diseases based on shared genes, given information retrieved from PsyGeNET. The user can compute the Jaccard Index using the function jaccardEstimation. This function accepts multiple inputs:

1. Using a list of genes of interest the function will compute the Jaccard Index between the set of genes and all the diseases in PsyGeNET.

```
genes_interest <- c("SLC6A4", "DRD2", "HTR1B", "PLP1", "TH", "DRD3")
ji1 <- jaccardEstimation(genes_interest, database = "ALL")

## Warning in singleInput.genes(diseases$diseases$‘gene-list‘$genes, psy, universe,
:  Jaccard Index for all diseases in PsyGeNET will be calculated.
```

2. Using a list of genes of interest and a list of diseases of interest, the function computes the Jaccard Index between the set of genes and each disease:

```
disease_interest <-
  c("delirium", "bipolar i disorder", "severe depression", "cocaine addiction")
ji2 <- jaccardEstimation(genes_interest, disease_interest, database = "ALL")
```

3. With a list of diseases of interest, the function will calculate the Jaccard Index between themselves:

```
ji3 <- jaccardEstimation(disease_interest, database = "ALL")
```

To determine if the association between two diseases as estimated by the Jaccard Index was statistically significant, we applied a bootstrap procedure to estimate the likelihood of obtaining a Jaccard Index greater than the one obtained for the association between the diseases by chance. In other words, we sampled at random gene sets of size n and p (n, p is the number of genes associated to disease 1 and 2, respectively) from a population of human disease genes obtained from DisGeNET [3]. These random gene sets (n and p) were then used to compute the Jaccard Index for diseases 1 and 2. This procedure was repeated 100 times. Then, we calculated the number of times we obtained a Jaccard Index for the random gene sets larger than the observed value of the Jaccard Index.

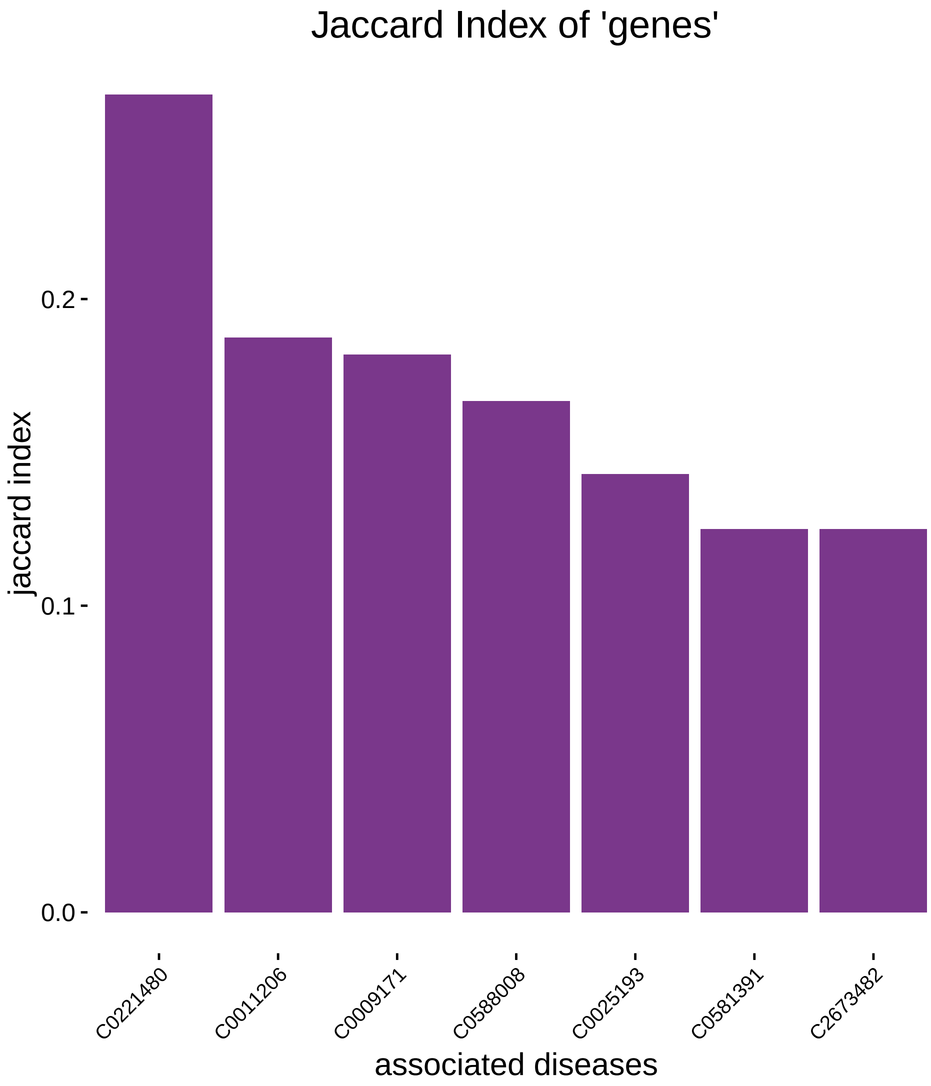
The raw results are stored in JaccardIndexPsy and can be obtained using the function extract. For example:

```
head(extract(ji1))

##   Disease1 Disease2 NGenes1 NGenes2 JaccardIndex pval
## 1    genes C0001973       6     279   0.01754386    0
## 2    genes C0005586       6     502   0.00984252    0
## 3    genes C0011581       6     276   0.01773050    0
## 4    genes C0525045       6     185   0.02617801    0
## 5    genes C1269683       6     254   0.01923077    0
## 6    genes C0011570       6     260   0.01879699    0

tail(extract(ji1))

##    Disease1 Disease2 NGenes1 NGenes2 JaccardIndex       pval
## 15    genes C0001969       6      33  0.025641026 0.02970297
## 37    genes C0349204       6      84  0.000000000 0.02970297
## 41    genes C0036337       6      21  0.000000000 0.02970297
## 52    genes C0036349       6      16  0.000000000 0.02970297
## 38    genes C0033975       6     109  0.000000000 0.05940594
## 36    genes C0036341       6     861  0.001153403 0.46534653
```

4.2 Plotting results: Jaccard Index

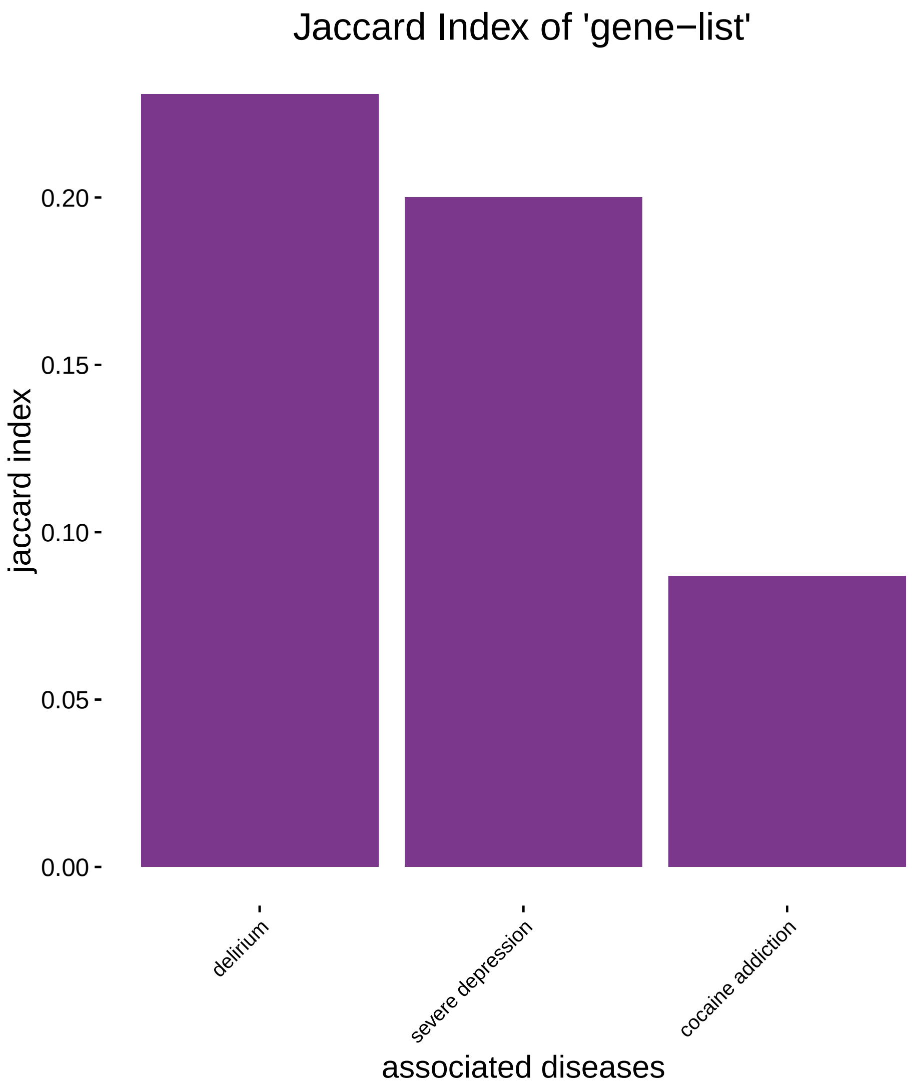
The plot of the result of a `jaccardEstimation` using a singe set of genes corresponds to a bar-plot of the Jaccard Index with each disease:

```
plot(ji1, cutOff = 0.1)
```



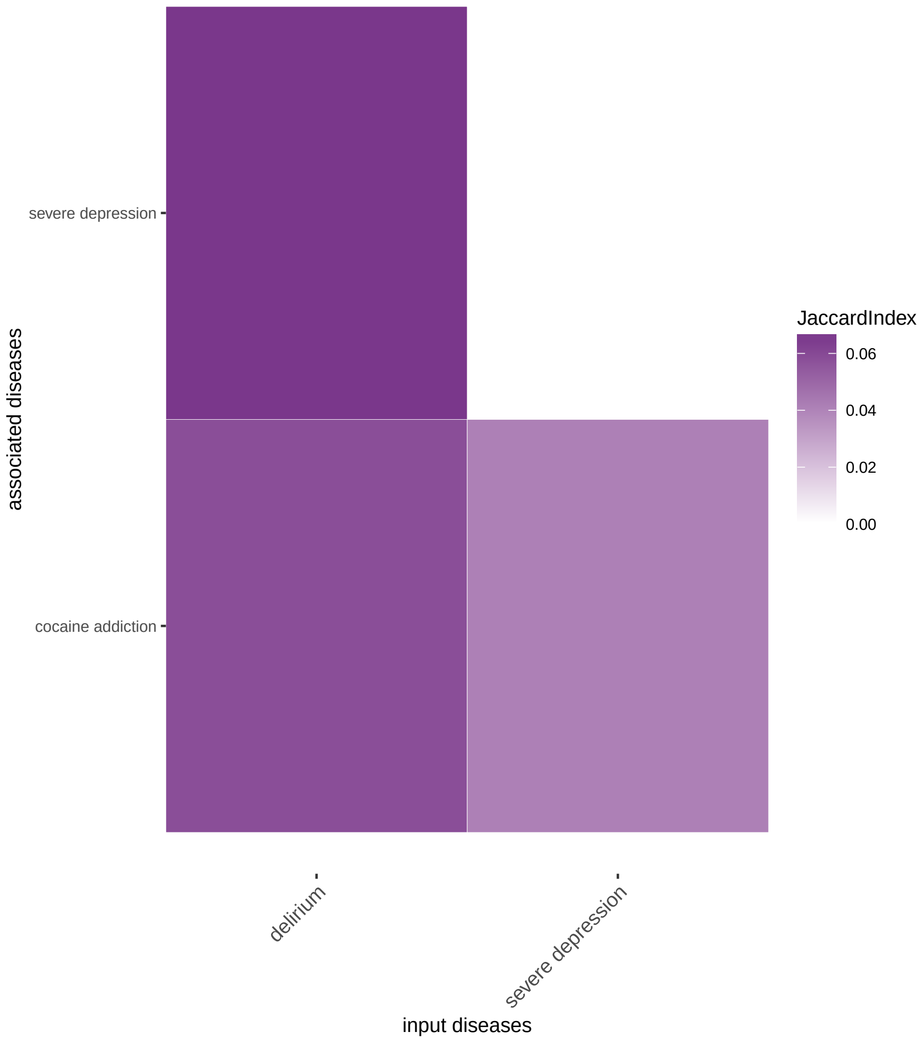
The previous bar-plot shows the Jaccard Index greater than 0.1 obtained from testing each diseases in PsyGeNET. When given a set of genes and a set of diseases, the resulting plot is equivalent:

```
plot(ji2)
```



The plot resulting from more than one disease is a heat-map with the given disease as X axis and all the diseases that share genes with them placed as Y axis. The intensity of the color represents the value of the Jaccard Index between, being the darker one the major Jaccard Index.

```
plot(ji3)
```



5 Warnings

`venn` and `heatmap` type arguments do not allow queries for single gene:

```
> plot( t1, type = "venn" )
==> Error: For this type of chart, a multiple gene query created with
 'psygenetGeneList' is required.

> plot( t1, type = "vennA" )
==> Error: For this type of chart, a multiple gene query created with
 'psygenetGeneList' is required.

> plot( t1, type = "heatmap" )
==> Error: For this type of chart, a multiple gene query created with
 'psygenetGeneList' is required.
```

References

[1] Alba Gutierrez-Sacristan; Solene Grosdidier; Olga Valverde; Marta Torrens; Alex Bravo; Janet Pinero; Ferran Sanz; Laura I. Furlong. **PsyGeNET: a knowledge platform on psychiatric disorders and their genes** Bioinformatics 2015 doi: 10.1093/bioinformatics/btv301

[2] Sullivan, Patrick F; Daly, Mark J; O'Donovan, Michael. **Genetic architectures of psychiatric disorders: the emerging picture and its implications** Nature reviews. Genetics (2012) vol. 13 (8) p. 537-51

[3] Janet Piñero, Núria Queralt-Rosinach, Àlex Bravo, Jordi Deu-Pons, Anna Bauer-Mehren, Martin Baron, Ferran Sanz, Laura I Furlong. **DisGeNET: a discovery platform for the dynamical exploration of human diseases and their genes** Database (2015) Vol. 2015: article ID bav028; doi:10.1093/database/bav028

[4] Bravo, À.; Piñero, J.; Queralt, N.; Rautschka, M.; Furlong, L.I. **Extraction of relations between genes and diseases from text and large-scale data analysis: implications for translational research** BMC Bioinformatics 2015, 16:55 doi:10.1186/s12859-015-0472-9

[5] Flint, Jonathan; Kendler, Kenneth S. **The genetics of major depression** Neuron (2014) Vol. 81 (3) p. 484-503

[6] Jens Treutlein, Sven Cichon et al. **Genome-wide association study of alcohol dependence**. Archives of general psychiatry (2009) vol.66(7) p.773 doi: 10.1001/archgenpsychiatry.2009.83.