SynaptomeDB: database for Synaptic proteome

Manual for querying SynaptomeDB

Oksana Sorokina, Anatoly Sorokin, J. Douglas Armstrong

17.10.2022

# Introduction

58 published synaptic proteomic datasets (2000-2022 years) that describe over 8,000 proteins were integrated and combined with direct protein-protein interactions and functional metadata to build a network resource.

The set includes 29 post synaptic proteome (PSP) studies (2000 to 2019) contributing a total of 5,560 mouse and human unique gene identifiers; 19 presynaptic studies (2004 to 2022) describe 2,772 unique human and mouse gene IDs, and 11 studies that span the whole synaptosome and report 7,198 unique genes.

*NB*: With the latest update of the database we have added 6 new studies: 1 presynaptic and 5 postsynaptic, new compsrtment (Synaptic Vesicle), and coding mutations for Autistic Spectral Disorder (ASD) and epilepsy (Epi).

To reconstruct protein-protein interaction (PPI) networks for the pre- and post-synaptic proteomes we used human PPI data filtered for the highest confidence direct and physical interactions from BioGRID, Intact and DIP. The resulting postsynaptic proteome (PSP) network contains 4,817 nodes and 27,788 edges in the Largest Connected Component (LCC). The presynaptic network is significantly smaller and comprises 2,221 nodes and 8,678 edges in the LCC.

The database includes: proteomic and interactomic data with supporting information on compartment, specie and brain region, GO function information for three species: mouse, rat and human, disease annotation for human (based on Human Disease Ontology (HDO) and GeneToModel table, which links certain synaptic proteins to existing computational models of synaptic plasticity and synaptic signal transduction

The original files are maintained at Ednburgh Datashare <https://doi.org/10.7488/ds/3017>. Updated database file could be found here: <https://doi.org/10.7488/ds/3771>

The dataset was described in the @Sorokina:2021hl.

# Overview of capabilities

suppressMessages(library(synaptome.db))  
suppressMessages(library(dplyr))  
library(ggplot2)  
library(pander)

## 1.Get information for a specific gene or gene set.

The dataset can be used to answer frequent questions such as “What is known about my favourite gene? Is it pre- or postsynaptic? Which brain region was it identified in?”, “Which publication it was reported in?” Information could be obtained by submitting gene EntrezID or Gene name

t <- getGeneInfoByEntrez(1742)   
pander(head(t))

Table continues below

| GeneID | Localisation | MGI | HumanEntrez | MouseEntrez | HumanName |
| --- | --- | --- | --- | --- | --- |
| 1 | Postsynaptic | MGI:1277959 | 1742 | 13385 | DLG4 |
| 1 | Postsynaptic | MGI:1277959 | 1742 | 13385 | DLG4 |
| 1 | Postsynaptic | MGI:1277959 | 1742 | 13385 | DLG4 |
| 1 | Postsynaptic | MGI:1277959 | 1742 | 13385 | DLG4 |
| 1 | Postsynaptic | MGI:1277959 | 1742 | 13385 | DLG4 |
| 1 | Postsynaptic | MGI:1277959 | 1742 | 13385 | DLG4 |

Table continues below

| MouseName | PaperPMID | Paper | Dataset | Year | SpeciesTaxID |
| --- | --- | --- | --- | --- | --- |
| Dlg4 | 10818142 | WALIKONIS\_2000 | FULL | 2000 | 10116 |
| Dlg4 | 10862698 | HUSI\_2000 | FULL | 2000 | 10090 |
| Dlg4 | 11895482 | SATON\_2002 | FULL | 2002 | 10090 |
| Dlg4 | 14532281 | LI\_2004 | FULL | 2004 | 10116 |
| Dlg4 | 14720225 | YOSHIMURA\_2004 | FULL | 2004 | 10116 |
| Dlg4 | 15020595 | PENG\_2002 | FULL | 2004 | 10116 |

| BrainRegion |
| --- |
| Forebrain |
| Forebrain |
| Forebrain |
| Forebrain |
| Forebrain |
| Forebrain |

t <- getGeneInfoByName("CASK")  
pander(head(t))

Table continues below

| GeneID | Localisation | MGI | HumanEntrez | MouseEntrez | HumanName |
| --- | --- | --- | --- | --- | --- |
| 409 | Postsynaptic | MGI:1309489 | 8573 | 12361 | CASK |
| 409 | Postsynaptic | MGI:1309489 | 8573 | 12361 | CASK |
| 409 | Postsynaptic | MGI:1309489 | 8573 | 12361 | CASK |
| 409 | Postsynaptic | MGI:1309489 | 8573 | 12361 | CASK |
| 409 | Postsynaptic | MGI:1309489 | 8573 | 12361 | CASK |
| 409 | Postsynaptic | MGI:1309489 | 8573 | 12361 | CASK |

Table continues below

| MouseName | PaperPMID | Paper | Dataset | Year | SpeciesTaxID |
| --- | --- | --- | --- | --- | --- |
| Cask | 15169875 | JORDAN\_2004 | FULL | 2004 | 10090 |
| Cask | 16635246 | COLLINS\_2006 | FULL | 2006 | 10090 |
| Cask | 17623647 | DOSEMESI\_2007 | FULL | 2007 | 10116 |
| Cask | 18056256 | TRINIDAD\_2008 | FULL | 2008 | 10090 |
| Cask | 18056256 | TRINIDAD\_2008 | FULL | 2008 | 10090 |
| Cask | 18056256 | TRINIDAD\_2008 | FULL | 2008 | 10090 |

| BrainRegion |
| --- |
| Brain |
| Forebrain |
| Cerebral cortex |
| Midbrain |
| Cerebellum |
| Hippocampus |

t <- getGeneInfoByName(c("CASK", "DLG2"))  
pander(head(t))

Table continues below

| GeneID | Localisation | MGI | HumanEntrez | MouseEntrez | HumanName |
| --- | --- | --- | --- | --- | --- |
| 6 | Postsynaptic | MGI:1344351 | 1740 | 23859 | DLG2 |
| 6 | Postsynaptic | MGI:1344351 | 1740 | 23859 | DLG2 |
| 6 | Postsynaptic | MGI:1344351 | 1740 | 23859 | DLG2 |
| 6 | Postsynaptic | MGI:1344351 | 1740 | 23859 | DLG2 |
| 6 | Postsynaptic | MGI:1344351 | 1740 | 23859 | DLG2 |
| 6 | Postsynaptic | MGI:1344351 | 1740 | 23859 | DLG2 |

Table continues below

| MouseName | PaperPMID | Paper | Dataset | Year | SpeciesTaxID |
| --- | --- | --- | --- | --- | --- |
| Dlg2 | 10862698 | HUSI\_2000 | FULL | 2000 | 10090 |
| Dlg2 | 11895482 | SATON\_2002 | FULL | 2002 | 10090 |
| Dlg2 | 14532281 | LI\_2004 | FULL | 2004 | 10116 |
| Dlg2 | 14720225 | YOSHIMURA\_2004 | FULL | 2004 | 10116 |
| Dlg2 | 15169875 | JORDAN\_2004 | FULL | 2004 | 10090 |
| Dlg2 | 15748150 | TRINIDAD\_2005 | FULL | 2005 | 10090 |

| BrainRegion |
| --- |
| Forebrain |
| Forebrain |
| Forebrain |
| Forebrain |
| Brain |
| Brain |

## 2.Get internal GeneIDs for the list of genes.

Obtaining Internal database GeneIDs is a useful intermediate step for more complex queries including those for building protein-protein interaction (PPI) networks for compartments and brain regions. Internal GeneID is specie-neutral and unique, which allows exact identification of the object of interest in case of redundancy (e.g. one Human genes matches on a few mouse ones, etc.)

t <- findGenesByEntrez(c(1742, 1741, 1739, 1740))  
pander(head(t))

Table continues below

| GeneID | MGI | HumanEntrez | MouseEntrez | RatEntrez | HumanName |
| --- | --- | --- | --- | --- | --- |
| 1 | MGI:1277959 | 1742 | 13385 | 29495 | DLG4 |
| 6 | MGI:1344351 | 1740 | 23859 | 64053 | DLG2 |
| 15 | MGI:1888986 | 1741 | 53310 | 58948 | DLG3 |
| 46 | MGI:107231 | 1739 | 13383 | 25252 | DLG1 |

| MouseName | RatName |
| --- | --- |
| Dlg4 | Dlg4 |
| Dlg2 | Dlg2 |
| Dlg3 | Dlg3 |
| Dlg1 | Dlg1 |

t <- findGenesByName(c("SRC", "SRCIN1", "FYN"))  
pander(head(t))

Table continues below

| GeneID | MGI | HumanEntrez | MouseEntrez | RatEntrez | HumanName |
| --- | --- | --- | --- | --- | --- |
| 48 | MGI:1933179 | 80725 | 56013 | 56029 | SRCIN1 |
| 585 | MGI:98397 | 6714 | 20779 | 83805 | SRC |
| 710 | MGI:95602 | 2534 | 14360 | 25150 | FYN |

| MouseName | RatName |
| --- | --- |
| Srcin1 | Srcin1 |
| Src | Src |
| Fyn | Fyn |

## 3.Get disease information for the gene set

Synaptic genes are annotated with disease information from Human Disease Ontology, where available. To get disease information one can submit the list of Human Entrez Is or Human genes names, it could be also the list of Internal GeneIDs if using getGeneDiseaseByIDs function

t <- getGeneDiseaseByName (c("CASK", "DLG2", "DLG1"))  
pander(head(t))

| HumanEntrez | HumanName | HDOID | Description |
| --- | --- | --- | --- |
| 1740 | DLG2 | DOID:936 | brain\_disease |
| 8573 | CASK | DOID:936 | brain\_disease |
| 1739 | DLG1 | DOID:331 | central\_nervous\_system\_disease |
| 1740 | DLG2 | DOID:331 | central\_nervous\_system\_disease |
| 8573 | CASK | DOID:331 | central\_nervous\_system\_disease |
| 1739 | DLG1 | DOID:863 | nervous\_system\_disease |

t <- getGeneDiseaseByEntres (c(8573, 1742, 1739))  
pander(head(t))

| HumanEntrez | HumanName | HDOID | Description |
| --- | --- | --- | --- |
| 8573 | CASK | DOID:936 | brain\_disease |
| 1739 | DLG1 | DOID:331 | central\_nervous\_system\_disease |
| 1742 | DLG4 | DOID:331 | central\_nervous\_system\_disease |
| 8573 | CASK | DOID:331 | central\_nervous\_system\_disease |
| 1739 | DLG1 | DOID:863 | nervous\_system\_disease |
| 1742 | DLG4 | DOID:863 | nervous\_system\_disease |

## 5.Get information about the studies, combined into dataset

One can obtain the overview of synaptic proteome papers combined into the database, which includes paper PMID, specie Tax ID, year of publication, subcellular localisation, brain region and number of proteins identified in the paper. This information may help to choose the specific study(ies) for further work.

p <- getPapers()  
pander(head(p))

Table continues below

| PaperPMID | SpeciesTaxID | Year | Name | Localisation | BrainRegion |
| --- | --- | --- | --- | --- | --- |
| 10818142 | 10116 | 2000 | WALIKONIS\_2000 | Postsynaptic | Forebrain |
| 10862698 | 10090 | 2000 | HUSI\_2000 | Postsynaptic | Forebrain |
| 11895482 | 10090 | 2002 | SATON\_2002 | Postsynaptic | Forebrain |
| 14532281 | 10116 | 2004 | LI\_2004 | Postsynaptic | Forebrain |
| 14720225 | 10116 | 2004 | YOSHIMURA\_2004 | Postsynaptic | Forebrain |
| 15007177 | 10116 | 2004 | BLONDEAU\_2004 | Presynaptic | Brain |

| Method | Ngenes |
| --- | --- |
| Shotgun | 29 |
| Shotgun | 77 |
| Shotgun | 45 |
| Shotgun | 138 |
| Shotgun | 436 |
| Shotgun | 209 |

## 6.Get the table of frequently identified proteins

It is also possible to obtain the list of proteins found in more than one study, for the whole synaptic proteome. For that, the user needs to provide a “count” value as a desired minimal number of identifications (e.g. 2 or more). The command returns the table with gene identifiers and “Npmid” column, which contains the number of studies where this gene was identified.

gp <- findGeneByPaperCnt(cnt = 2)#find all proteins in synaptic proteome identified 2 times or more  
#> Warning in result\_fetch(res@ptr, n = n): Column `RatEntrez`: mixed type, first  
#> seen values of type integer, coercing other values of type string  
pander(head(gp))

Table continues below

| GeneID | MGI | HumanEntrez | MouseEntrez | RatEntrez | HumanName |
| --- | --- | --- | --- | --- | --- |
| 1 | MGI:1277959 | 1742 | 13385 | 29495 | DLG4 |
| 2 | MGI:88256 | 815 | 12322 | 25400 | CAMK2A |
| 3 | MGI:96568 | 9118 | 226180 | 24503 | INA |
| 4 | MGI:98388 | 6711 | 20742 | 305614 | SPTBN1 |
| 5 | MGI:88257 | 816 | 12323 | 24245 | CAMK2B |
| 6 | MGI:1344351 | 1740 | 23859 | 64053 | DLG2 |

| MouseName | RatName | Npmid |
| --- | --- | --- |
| Dlg4 | Dlg4 | 48 |
| Camk2a | Camk2a | 54 |
| Ina | Ina | 50 |
| Sptbn1 | Sptbn1 | 45 |
| Camk2b | Camk2b | 46 |
| Dlg2 | Dlg2 | 43 |

## 7.Get the table of proteins identified in specific studies

Following section 5, when the information for all considered proteomic studies was obtained, user can select specific study(ies) by PMID and get the proteins identified in those studies. By providing “count” value user can extract either all proteins from specified studies (count = 1), or just frequently found ones (count >=2). As above, the command returns the table with gene identifiers with “Npmid” column, which contains the number of studies where this protein was identified

spg <- findGeneByPapers(p$PaperPMID[1:5], cnt = 1)  
pander(head(spg))

Table continues below

| GeneID | MGI | HumanEntrez | MouseEntrez | RatEntrez | HumanName |
| --- | --- | --- | --- | --- | --- |
| 1 | MGI:1277959 | 1742 | 13385 | 29495 | DLG4 |
| 2 | MGI:88256 | 815 | 12322 | 25400 | CAMK2A |
| 3 | MGI:96568 | 9118 | 226180 | 24503 | INA |
| 4 | MGI:98388 | 6711 | 20742 | 305614 | SPTBN1 |
| 5 | MGI:88257 | 816 | 12323 | 24245 | CAMK2B |
| 6 | MGI:1344351 | 1740 | 23859 | 64053 | DLG2 |

| MouseName | RatName | Npmid |
| --- | --- | --- |
| Dlg4 | Dlg4 | 5 |
| Camk2a | Camk2a | 5 |
| Ina | Ina | 5 |
| Sptbn1 | Sptbn1 | 4 |
| Camk2b | Camk2b | 4 |
| Dlg2 | Dlg2 | 4 |

## 8.Get the table of proteins frequently identified in specific compartment

Most of the times, user is interested in the specific compartment rather then in total synaptic proteome. To help identify the genes most probably residenting in the specific compartment and exclude possible contaminants, findGeneByCompartmentPaperCnt function provides the table of proteins found “cnt” or more times in different compartment-paper pairs.

gcp <- findGeneByCompartmentPaperCnt(cnt = 2)  
#> Warning in result\_fetch(res@ptr, n = n): Column `RatEntrez`: mixed type, first  
#> seen values of type integer, coercing other values of type string  
pander(head(gcp))

Table continues below

| GeneID | MGI | HumanEntrez | MouseEntrez | RatEntrez | HumanName |
| --- | --- | --- | --- | --- | --- |
| 1 | MGI:1277959 | 1742 | 13385 | 29495 | DLG4 |
| 1 | MGI:1277959 | 1742 | 13385 | 29495 | DLG4 |
| 1 | MGI:1277959 | 1742 | 13385 | 29495 | DLG4 |
| 1 | MGI:1277959 | 1742 | 13385 | 29495 | DLG4 |
| 2 | MGI:88256 | 815 | 12322 | 25400 | CAMK2A |
| 2 | MGI:88256 | 815 | 12322 | 25400 | CAMK2A |

| MouseName | RatName | Localisation | Npmid |
| --- | --- | --- | --- |
| Dlg4 | Dlg4 | Postsynaptic | 29 |
| Dlg4 | Dlg4 | Presynaptic | 4 |
| Dlg4 | Dlg4 | Synaptosome | 16 |
| Dlg4 | Dlg4 | Synaptic\_Vesicle | 3 |
| Camk2a | Camk2a | Postsynaptic | 28 |
| Camk2a | Camk2a | Presynaptic | 13 |

Now user can select the specific compartment and proceed working with obtained list of frequently found proteins

presgp <- gcp[gcp$Localisation == "Presynaptic",]  
dim(presgp)  
#> [1] 1542 10  
pander(head(presgp))

Table continues below

| GeneID | MGI | HumanEntrez | MouseEntrez | RatEntrez | HumanName |
| --- | --- | --- | --- | --- | --- |
| 1 | MGI:1277959 | 1742 | 13385 | 29495 | DLG4 |
| 2 | MGI:88256 | 815 | 12322 | 25400 | CAMK2A |
| 3 | MGI:96568 | 9118 | 226180 | 24503 | INA |
| 4 | MGI:98388 | 6711 | 20742 | 305614 | SPTBN1 |
| 5 | MGI:88257 | 816 | 12323 | 24245 | CAMK2B |
| 6 | MGI:1344351 | 1740 | 23859 | 64053 | DLG2 |

| MouseName | RatName | Localisation | Npmid |
| --- | --- | --- | --- |
| Dlg4 | Dlg4 | Presynaptic | 4 |
| Camk2a | Camk2a | Presynaptic | 13 |
| Ina | Ina | Presynaptic | 12 |
| Sptbn1 | Sptbn1 | Presynaptic | 8 |
| Camk2b | Camk2b | Presynaptic | 8 |
| Dlg2 | Dlg2 | Presynaptic | 3 |

## 9.Get PPI interactions for my list of genes

Custom Protein-protein interactions based on bespoke subsets of molecules could be extracted in two general ways: “induced” and “limited”. In the first case, the command will return all possible interactors for the genes within the whole interactome. In the second case it will return only interactions between the genes of interest. PPIs could be obtained by submitting list of EntrezIDs or gene names, or Internal IDs - in all cases the interactions will be returned as a list of interacting pairs of Intenal GeneIDs.

t <- getPPIbyName(  
 c("CASK", "DLG4", "GRIN2A", "GRIN2B","GRIN1"),   
 type = "limited")  
pander(head(t))

| A | B |
| --- | --- |
| 38 | 1 |
| 7 | 1 |
| 1 | 7 |
| 1 | 38 |
| 1 | 9 |
| 9 | 1 |

t <- getPPIbyEntrez(c(1739, 1740, 1742, 1741), type='induced')  
pander(head(t))

| A | B |
| --- | --- |
| 1 | 2871 |
| 6 | 2871 |
| 15 | 2871 |
| 1 | 617 |
| 1 | 30 |
| 156 | 1 |

#obtain PPIs for the list of frequently found genes in presynaptc compartment  
t <- getPPIbyEntrez(presgp$HumanEntrez, type='induced')   
pander(head(t))

| A | B |
| --- | --- |
| 365 | 148 |
| 1048 | 148 |
| 52 | 365 |
| 52 | 1048 |
| 321 | 1048 |
| 321 | 365 |

## 10.Get the molecular structure of synaptic compartment

Three main synaptic compartments considered in the database are “presynaptic”, “postsynaptic” and “synaptosome”. Genes are classified to compartments based on respective publications, so that each gene can belong to one or two, or even three compartments. The full list of genes for specific compartment could be obtained with command getAllGenes4Compartment, which returns the table with main gene identifiers, like internal GeneIDs, MGI ID, Human Entrez ID, Human Gene Name, Mouse Entrez ID, Mouse Gene Name, Rat Entrez ID, Rat Gene Name.

If you need to check which genes of your list belong to specific compartment, you can use getGenes4Compartment command, which will select from your list only genes associated with specific compartment. To obtain the PPI network for compartment one has to submit the list of Internal GeneIDs obtained with previous commands.

#getting the list of compartment  
comp <- getCompartments()  
pander(comp)

| ID | Name | Description |
| --- | --- | --- |
| 1 | Postsynaptic | Postsynaptic |
| 2 | Presynaptic | Presynaptic |
| 3 | Synaptosome | Synaptosome |
| 4 | Synaptic\_Vesicle | Synaptic\_Vesicle |

#getting all genes for postsynaptic compartment  
gns <- getAllGenes4Compartment(compartmentID = 1)   
pander(head(gns))

Table continues below

| GeneID | MGI | HumanEntrez | MouseEntrez | RatEntrez | HumanName |
| --- | --- | --- | --- | --- | --- |
| 1 | MGI:1277959 | 1742 | 13385 | 29495 | DLG4 |
| 2 | MGI:88256 | 815 | 12322 | 25400 | CAMK2A |
| 3 | MGI:96568 | 9118 | 226180 | 24503 | INA |
| 4 | MGI:98388 | 6711 | 20742 | 305614 | SPTBN1 |
| 5 | MGI:88257 | 816 | 12323 | 24245 | CAMK2B |
| 6 | MGI:1344351 | 1740 | 23859 | 64053 | DLG2 |

| MouseName | RatName |
| --- | --- |
| Dlg4 | Dlg4 |
| Camk2a | Camk2a |
| Ina | Ina |
| Sptbn1 | Sptbn1 |
| Camk2b | Camk2b |
| Dlg2 | Dlg2 |

#getting full PPI network for postsynaptic compartment  
ppi <- getPPIbyIDs4Compartment(gns$GeneID,compartmentID =1, type = "induced")  
pander(head(ppi))

| A | B |
| --- | --- |
| 365 | 148 |
| 1048 | 148 |
| 52 | 365 |
| 52 | 1048 |
| 321 | 1048 |
| 321 | 365 |

## 11.Get the molecular structure of the brain region.

Three are 12 brain regions considered in the database based on respective publications, so that each gene can belong to the single or to the several brain regions. Brain regions differ between species, and specie brain region information is not 100% covered in the database(e.g. we don’t have yet studies for Human Striatum, but do have for Mouse and Rat), that’s why when querying the database for brain region information you will need to specify the specie. The full list of genes for specific region could be obtained wuth command getAllGenes4BrainRegion, which returns the table with main gene identifiers, like internal Gene IDs, MGI ID, Human Entrez ID, Human Gene Name, Mouse Entrez ID, Mouse Gene Name, Rat Entrez ID, Rat Gene Name.

If you need to check which genes of your list were identified in specific region, you can use getGenes4BrainRegion command, which will select only genes associated with specific region from your list.

To obtain the PPI network for brain region you need to submit the list of Internal GeneIDs obtained with previous commands.

#getting the full list of brain regions  
reg <- getBrainRegions()  
pander(reg)

| ID | Name | Description | InterlexID | ParentID |
| --- | --- | --- | --- | --- |
| 1 | Brain | Whole brain | ILX:0101431 | 1 |
| 2 | Forebrain | Whole forebrain | ILX:0104355 | 1 |
| 3 | Midbrain | Midbrain | ILX:0106935 | 1 |
| 4 | Cerebellum | Cerebellum | ILX:0101963 | 1 |
| 5 | Telencephalon | Telencephalon | ILX:0111558 | 2 |
| 6 | Hypothalamus | Hypothalamus | ILX:0105177 | 2 |
| 7 | Hippocampus | Hippocampus | ILX:0105021 | 5 |
| 8 | Striatum | Striatum | ILX:0111098 | 5 |
| 9 | Cerebral cortex | Neocortex | ILX:0101978 | 5 |
| 10 | Frontal lobe | Frontal lobe/frontal cortex | ILX:0104451 | 9 |
| 11 | Occipital lobe | Occipital lobe | ILX:0107883 | 9 |
| 12 | Temporal lobe | Temporal lobe | ILX:0111590 | 9 |
| 13 | Parietal lobe | Parietal lobe | ILX:0108534 | 9 |
| 14 | Prefrontal cortex | Prefrontal cortex | ILX:0109209 | 10 |
| 15 | Motor cortex | Motor cortex | ILX:0107119 | 10 |
| 16 | Visual cortex | Visual cortex | ILX:0112513 | 11 |
| 17 | Medial cortex | Medial cortex | ILX:0106634 | 9 |
| 18 | Caudal cortex | Caudal cortex | NA | 9 |

#getting all genes for mouse Striatum  
gns <- getAllGenes4BrainRegion(brainRegion = "Striatum",taxID = 10090)  
pander(head(gns))

Table continues below

| GeneID | Localisation | MGI | HumanEntrez | MouseEntrez | HumanName |
| --- | --- | --- | --- | --- | --- |
| 1 | Postsynaptic | MGI:1277959 | 1742 | 13385 | DLG4 |
| 2 | Postsynaptic | MGI:88256 | 815 | 12322 | CAMK2A |
| 3 | Postsynaptic | MGI:96568 | 9118 | 226180 | INA |
| 4 | Postsynaptic | MGI:98388 | 6711 | 20742 | SPTBN1 |
| 6 | Postsynaptic | MGI:1344351 | 1740 | 23859 | DLG2 |
| 7 | Postsynaptic | MGI:95821 | 2904 | 14812 | GRIN2B |

| MouseName | PMID | Paper | Year | SpeciesTaxID | BrainRegion |
| --- | --- | --- | --- | --- | --- |
| Dlg4 | 30071621 | ROY\_2018 | 2018 | 10090 | Striatum |
| Camk2a | 30071621 | ROY\_2018 | 2018 | 10090 | Striatum |
| Ina | 30071621 | ROY\_2018 | 2018 | 10090 | Striatum |
| Sptbn1 | 30071621 | ROY\_2018 | 2018 | 10090 | Striatum |
| Dlg2 | 30071621 | ROY\_2018 | 2018 | 10090 | Striatum |
| Grin2b | 30071621 | ROY\_2018 | 2018 | 10090 | Striatum |

#getting full PPI network for postsynaptic compartment  
ppi <- getPPIbyIDs4BrainRegion(  
 gns$GeneID, brainRegion = "Striatum",   
 taxID = 10090, type = "limited")  
pander(head(ppi))

| A | B |
| --- | --- |
| 365 | 148 |
| 1048 | 148 |
| 52 | 365 |
| 52 | 1048 |
| 321 | 1048 |
| 321 | 365 |

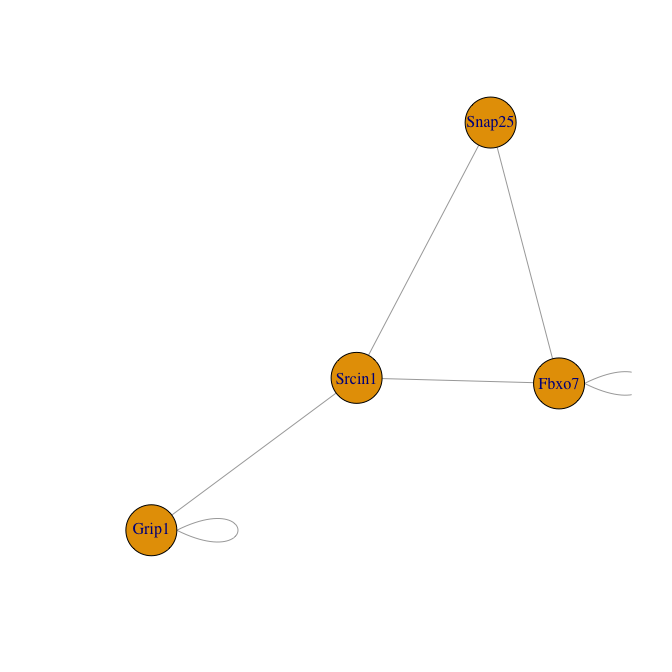
## 12.Checking third-party list against Synaptic Proteome db

listG<-findGenesByEntrez(1:250) #check whic genes from 250 random EntrezIds are in the database  
dim(listG)  
#> [1] 124 8  
head(listG)  
#> # A tibble: 6 × 8  
#> GeneID MGI HumanEntrez MouseEntrez RatEntrez HumanName Mouse…¹ RatName  
#> <int> <chr> <int> <int> <int> <chr> <chr> <chr>   
#> 1 34 MGI:87994 226 11674 24189 ALDOA Aldoa Aldoa   
#> 2 35 MGI:2137706 87 109711 81634 ACTN1 Actn1 Actn1   
#> 3 39 MGI:101921 160 11771 308578 AP2A1 Ap2a1 Ap2a1   
#> 4 54 MGI:87918 118 11518 24170 ADD1 Add1 Add1   
#> 5 85 MGI:87919 119 11519 24171 ADD2 Add2 Add2   
#> 6 97 MGI:1890773 81 60595 63836 ACTN4 Actn4 Actn4   
#> # … with abbreviated variable name ¹​MouseName  
  
getCompartments()  
#> # A tibble: 4 × 3  
#> ID Name Description   
#> <int> <chr> <chr>   
#> 1 1 Postsynaptic Postsynaptic   
#> 2 2 Presynaptic Presynaptic   
#> 3 3 Synaptosome Synaptosome   
#> 4 4 Synaptic\_Vesicle Synaptic\_Vesicle  
presG <- getGenes4Compartment(listG$GeneID, 2) #check which genes from subset identified as synaptic are presynaptic  
dim(presG)  
#> [1] 67 8  
head(presG)  
#> # A tibble: 6 × 8  
#> GeneID MGI HumanEntrez MouseEntrez RatEntrez HumanName Mouse…¹ RatName  
#> <int> <chr> <int> <int> <int> <chr> <chr> <chr>   
#> 1 34 MGI:87994 226 11674 24189 ALDOA Aldoa Aldoa   
#> 2 35 MGI:2137706 87 109711 81634 ACTN1 Actn1 Actn1   
#> 3 39 MGI:101921 160 11771 308578 AP2A1 Ap2a1 Ap2a1   
#> 4 54 MGI:87918 118 11518 24170 ADD1 Add1 Add1   
#> 5 85 MGI:87919 119 11519 24171 ADD2 Add2 Add2   
#> 6 97 MGI:1890773 81 60595 63836 ACTN4 Actn4 Actn4   
#> # … with abbreviated variable name ¹​MouseName  
  
getBrainRegions()  
#> # A tibble: 18 × 5  
#> ID Name Description InterlexID ParentID  
#> <int> <chr> <chr> <chr> <int>  
#> 1 1 Brain Whole brain ILX:0101431 1  
#> 2 2 Forebrain Whole forebrain ILX:0104355 1  
#> 3 3 Midbrain Midbrain ILX:0106935 1  
#> 4 4 Cerebellum Cerebellum ILX:0101963 1  
#> 5 5 Telencephalon Telencephalon ILX:0111558 2  
#> 6 6 Hypothalamus Hypothalamus ILX:0105177 2  
#> 7 7 Hippocampus Hippocampus ILX:0105021 5  
#> 8 8 Striatum Striatum ILX:0111098 5  
#> 9 9 Cerebral cortex Neocortex ILX:0101978 5  
#> 10 10 Frontal lobe Frontal lobe/frontal cortex ILX:0104451 9  
#> 11 11 Occipital lobe Occipital lobe ILX:0107883 9  
#> 12 12 Temporal lobe Temporal lobe ILX:0111590 9  
#> 13 13 Parietal lobe Parietal lobe ILX:0108534 9  
#> 14 14 Prefrontal cortex Prefrontal cortex ILX:0109209 10  
#> 15 15 Motor cortex Motor cortex ILX:0107119 10  
#> 16 16 Visual cortex Visual cortex ILX:0112513 11  
#> 17 17 Medial cortex Medial cortex ILX:0106634 9  
#> 18 18 Caudal cortex Caudal cortex <NA> 9  
listR <- getGenes4BrainRegion(listG$GeneID, brainRegion = "Cerebellum", taxID = 10090) #check which genes from subset identified as synaptic are found in human cerebellum  
dim(listR)  
#> [1] 186 12  
head(listR)  
#> # A tibble: 6 × 12  
#> GeneID Localisation MGI Human…¹ Mouse…² Human…³ Mouse…⁴ PMID Paper Year  
#> <int> <chr> <chr> <int> <int> <chr> <chr> <int> <chr> <int>  
#> 1 34 Postsynaptic MGI:87… 226 11674 ALDOA Aldoa 1.81e7 TRIN… 2008  
#> 2 34 Postsynaptic MGI:87… 226 11674 ALDOA Aldoa 3.01e7 ROY\_… 2018  
#> 3 34 Synaptosome MGI:87… 226 11674 ALDOA Aldoa 3.20e7 DIST… 2020  
#> 4 34 Synaptosome MGI:87… 226 11674 ALDOA Aldoa 3.21e7 GONZ… 2020  
#> 5 35 Postsynaptic MGI:21… 87 109711 ACTN1 Actn1 1.81e7 TRIN… 2008  
#> 6 35 Postsynaptic MGI:21… 87 109711 ACTN1 Actn1 3.01e7 ROY\_… 2018  
#> # … with 2 more variables: SpeciesTaxID <int>, BrainRegion <chr>, and  
#> # abbreviated variable names ¹​HumanEntrez, ²​MouseEntrez, ³​HumanName,  
#> # ⁴​MouseName

## 13.Visualisatiion of PPI network with Igraph.

Combine information from PPI data.frame obtained with functions like getPPIbyName, getPPIbyEntrez, getPPIbyIDs4Compartment or getPPIbyIDs4BrainRegion with information about genes obtained from getGenesByID to make interpretable undirected PPI graph in igraph format. In this format network could be further analysed and visualized by algorithms from igraph package.

library(igraph)  
#>   
#> Attaching package: 'igraph'  
#> The following objects are masked from 'package:dplyr':  
#>   
#> as\_data\_frame, groups, union  
#> The following objects are masked from 'package:BiocGenerics':  
#>   
#> normalize, path, union  
#> The following objects are masked from 'package:stats':  
#>   
#> decompose, spectrum  
#> The following object is masked from 'package:base':  
#>   
#> union  
g<-getIGraphFromPPI(  
 getPPIbyIDs(c(48, 129, 975, 4422, 5715, 5835), type='lim'))  
plot(g,vertex.label=V(g)$RatName,vertex.size=25)



## 14.Export of PPI network as a table.

If Igraph is not an option, the PPI network could be exported as an interpretible table to be processed with other tools, e.g. Cytoscape,etc.

tbl<-getTableFromPPI(getPPIbyIDs(c(48, 585, 710), type='limited'))  
tbl  
#> # A tibble: 2 × 16  
#> A B MGI.A HumanEnt…¹ Mouse…² RatEn…³ Human…⁴ Mouse…⁵ RatNa…⁶ MGI.B  
#> <int> <int> <chr> <int> <int> <int> <chr> <chr> <chr> <chr>  
#> 1 710 710 MGI:95602 2534 14360 25150 FYN Fyn Fyn MGI:…  
#> 2 585 585 MGI:98397 6714 20779 83805 SRC Src Src MGI:…  
#> # … with 6 more variables: HumanEntrez.B <int>, MouseEntrez.B <int>,  
#> # RatEntrez.B <int>, HumanName.B <chr>, MouseName.B <chr>, RatName.B <chr>,  
#> # and abbreviated variable names ¹​HumanEntrez.A, ²​MouseEntrez.A,  
#> # ³​RatEntrez.A, ⁴​HumanName.A, ⁵​MouseName.A, ⁶​RatName.A

# References

# Appendix

## Versions

### Session Info

|  |  |
| --- | --- |
| **version** | R version 4.2.1 (2022-06-23) |
| **os** | macOS Big Sur … 10.16 |
| **system** | x86\_64, darwin17.0 |
| **ui** | X11 |
| **language** | (EN) |
| **collate** | en\_US.UTF-8 |
| **ctype** | en\_US.UTF-8 |
| **tz** | Asia/Tokyo |
| **date** | 2022-10-17 |
| **pandoc** | 2.19 @ /usr/local/bin/ (via rmarkdown) |

Table continues below

|  | package | ondiskversion |
| --- | --- | --- |
| **AnnotationDbi** | AnnotationDbi | 1.59.1 |
| **AnnotationHub** | AnnotationHub | 3.5.2 |
| **assertthat** | assertthat | 0.2.1 |
| **Biobase** | Biobase | 2.57.1 |
| **BiocFileCache** | BiocFileCache | 2.5.2 |
| **BiocGenerics** | BiocGenerics | 0.43.4 |
| **BiocManager** | BiocManager | 1.30.18 |
| **BiocVersion** | BiocVersion | 3.16.0 |
| **Biostrings** | Biostrings | 2.65.6 |
| **bit** | bit | 4.0.4 |
| **bit64** | bit64 | 4.0.5 |
| **bitops** | bitops | 1.0.7 |
| **blob** | blob | 1.2.3 |
| **cachem** | cachem | 1.0.6 |
| **callr** | callr | 3.7.2 |
| **cli** | cli | 3.4.1 |
| **colorspace** | colorspace | 2.0.3 |
| **crayon** | crayon | 1.5.2 |
| **curl** | curl | 4.3.3 |
| **DBI** | DBI | 1.1.3 |
| **dbplyr** | dbplyr | 2.2.1 |
| **devtools** | devtools | 2.4.5 |
| **digest** | digest | 0.6.29 |
| **dplyr** | dplyr | 1.0.10 |
| **ellipsis** | ellipsis | 0.3.2 |
| **evaluate** | evaluate | 0.17 |
| **fansi** | fansi | 1.0.3 |
| **fastmap** | fastmap | 1.1.0 |
| **filelock** | filelock | 1.0.2 |
| **fs** | fs | 1.5.2 |
| **generics** | generics | 0.1.3 |
| **GenomeInfoDb** | GenomeInfoDb | 1.33.7 |
| **GenomeInfoDbData** | GenomeInfoDbData | 1.2.9 |
| **ggplot2** | ggplot2 | 3.3.6 |
| **glue** | glue | 1.6.2 |
| **gtable** | gtable | 0.3.1 |
| **htmltools** | htmltools | 0.5.3 |
| **htmlwidgets** | htmlwidgets | 1.5.4 |
| **httpuv** | httpuv | 1.6.6 |
| **httr** | httr | 1.4.4 |
| **igraph** | igraph | 1.3.5 |
| **interactiveDisplayBase** | interactiveDisplayBase | 1.35.0 |
| **IRanges** | IRanges | 2.31.2 |
| **KEGGREST** | KEGGREST | 1.37.3 |
| **knitr** | knitr | 1.40 |
| **later** | later | 1.3.0 |
| **lifecycle** | lifecycle | 1.0.3 |
| **magrittr** | magrittr | 2.0.3 |
| **memoise** | memoise | 2.0.1 |
| **mime** | mime | 0.12 |
| **miniUI** | miniUI | 0.1.1.1 |
| **munsell** | munsell | 0.5.0 |
| **pander** | pander | 0.6.5 |
| **pillar** | pillar | 1.8.1 |
| **pkgbuild** | pkgbuild | 1.3.1 |
| **pkgconfig** | pkgconfig | 2.0.3 |
| **pkgload** | pkgload | 1.3.0 |
| **png** | png | 0.1.7 |
| **prettyunits** | prettyunits | 1.1.1 |
| **processx** | processx | 3.7.0 |
| **profvis** | profvis | 0.3.7 |
| **promises** | promises | 1.2.0.1 |
| **ps** | ps | 1.7.1 |
| **purrr** | purrr | 0.3.5 |
| **R6** | R6 | 2.5.1 |
| **rappdirs** | rappdirs | 0.3.3 |
| **rbibutils** | rbibutils | 2.2.9 |
| **Rcpp** | Rcpp | 1.0.9 |
| **RCurl** | RCurl | 1.98.1.9 |
| **Rdpack** | Rdpack | 2.4 |
| **remotes** | remotes | 2.4.2 |
| **rlang** | rlang | 1.0.6 |
| **rmarkdown** | rmarkdown | 2.17 |
| **RSQLite** | RSQLite | 2.2.18 |
| **rstudioapi** | rstudioapi | 0.14 |
| **S4Vectors** | S4Vectors | 0.35.4 |
| **scales** | scales | 1.2.1 |
| **sessioninfo** | sessioninfo | 1.2.2 |
| **shiny** | shiny | 1.7.2 |
| **stringi** | stringi | 1.7.8 |
| **stringr** | stringr | 1.4.1 |
| **synaptome.data** | synaptome.data | 0.99.3 |
| **synaptome.ldb** | synaptome.ldb | 0.99.9 |
| **tibble** | tibble | 3.1.8 |
| **tidyselect** | tidyselect | 1.2.0 |
| **urlchecker** | urlchecker | 1.0.1 |
| **usethis** | usethis | 2.1.6 |
| **utf8** | utf8 | 1.2.2 |
| **vctrs** | vctrs | 0.4.2 |
| **withr** | withr | 2.5.0 |
| **xfun** | xfun | 0.33 |
| **xtable** | xtable | 1.8.4 |
| **XVector** | XVector | 0.37.1 |
| **yaml** | yaml | 2.3.5 |
| **zlibbioc** | zlibbioc | 1.43.0 |

Table continues below

|  | loadedversion | attached | is\_base | date |
| --- | --- | --- | --- | --- |
| **AnnotationDbi** | 1.59.1 | FALSE | FALSE | 2022-05-19 |
| **AnnotationHub** | 3.5.2 | TRUE | FALSE | 2022-09-29 |
| **assertthat** | 0.2.1 | FALSE | FALSE | 2019-03-21 |
| **Biobase** | 2.57.1 | FALSE | FALSE | 2022-05-19 |
| **BiocFileCache** | 2.5.2 | TRUE | FALSE | 2022-10-06 |
| **BiocGenerics** | 0.43.4 | TRUE | FALSE | 2022-09-11 |
| **BiocManager** | 1.30.18 | FALSE | FALSE | 2022-05-18 |
| **BiocVersion** | 3.16.0 | FALSE | FALSE | 2022-05-05 |
| **Biostrings** | 2.65.6 | FALSE | FALSE | 2022-09-09 |
| **bit** | 4.0.4 | FALSE | FALSE | 2020-08-04 |
| **bit64** | 4.0.5 | FALSE | FALSE | 2020-08-30 |
| **bitops** | 1.0-7 | FALSE | FALSE | 2021-04-24 |
| **blob** | 1.2.3 | FALSE | FALSE | 2022-04-10 |
| **cachem** | 1.0.6 | FALSE | FALSE | 2021-08-19 |
| **callr** | 3.7.2 | FALSE | FALSE | 2022-08-22 |
| **cli** | 3.4.1 | FALSE | FALSE | 2022-09-23 |
| **colorspace** | 2.0-3 | FALSE | FALSE | 2022-02-21 |
| **crayon** | 1.5.2 | FALSE | FALSE | 2022-09-29 |
| **curl** | 4.3.3 | FALSE | FALSE | 2022-10-06 |
| **DBI** | 1.1.3 | FALSE | FALSE | 2022-06-18 |
| **dbplyr** | 2.2.1 | TRUE | FALSE | 2022-06-27 |
| **devtools** | 2.4.5 | FALSE | FALSE | 2022-10-11 |
| **digest** | 0.6.29 | FALSE | FALSE | 2021-12-01 |
| **dplyr** | 1.0.10 | TRUE | FALSE | 2022-09-01 |
| **ellipsis** | 0.3.2 | FALSE | FALSE | 2021-04-29 |
| **evaluate** | 0.17 | FALSE | FALSE | 2022-10-07 |
| **fansi** | 1.0.3 | FALSE | FALSE | 2022-03-24 |
| **fastmap** | 1.1.0 | FALSE | FALSE | 2021-01-25 |
| **filelock** | 1.0.2 | FALSE | FALSE | 2018-10-05 |
| **fs** | 1.5.2 | FALSE | FALSE | 2021-12-08 |
| **generics** | 0.1.3 | FALSE | FALSE | 2022-07-05 |
| **GenomeInfoDb** | 1.33.7 | FALSE | FALSE | 2022-09-07 |
| **GenomeInfoDbData** | 1.2.9 | FALSE | FALSE | 2022-10-04 |
| **ggplot2** | 3.3.6 | TRUE | FALSE | 2022-05-03 |
| **glue** | 1.6.2 | FALSE | FALSE | 2022-02-24 |
| **gtable** | 0.3.1 | FALSE | FALSE | 2022-09-01 |
| **htmltools** | 0.5.3 | FALSE | FALSE | 2022-07-18 |
| **htmlwidgets** | 1.5.4 | FALSE | FALSE | 2021-09-08 |
| **httpuv** | 1.6.6 | FALSE | FALSE | 2022-09-08 |
| **httr** | 1.4.4 | FALSE | FALSE | 2022-08-17 |
| **igraph** | 1.3.5 | TRUE | FALSE | 2022-09-22 |
| **interactiveDisplayBase** | 1.35.0 | FALSE | FALSE | 2022-05-05 |
| **IRanges** | 2.31.2 | FALSE | FALSE | 2022-08-18 |
| **KEGGREST** | 1.37.3 | FALSE | FALSE | 2022-07-10 |
| **knitr** | 1.40 | TRUE | FALSE | 2022-08-24 |
| **later** | 1.3.0 | FALSE | FALSE | 2021-08-18 |
| **lifecycle** | 1.0.3 | FALSE | FALSE | 2022-10-07 |
| **magrittr** | 2.0.3 | FALSE | FALSE | 2022-03-30 |
| **memoise** | 2.0.1 | FALSE | FALSE | 2021-11-26 |
| **mime** | 0.12 | FALSE | FALSE | 2021-09-28 |
| **miniUI** | 0.1.1.1 | FALSE | FALSE | 2018-05-18 |
| **munsell** | 0.5.0 | FALSE | FALSE | 2018-06-12 |
| **pander** | 0.6.5 | TRUE | FALSE | 2022-03-18 |
| **pillar** | 1.8.1 | FALSE | FALSE | 2022-08-19 |
| **pkgbuild** | 1.3.1 | FALSE | FALSE | 2021-12-20 |
| **pkgconfig** | 2.0.3 | FALSE | FALSE | 2019-09-22 |
| **pkgload** | 1.3.0 | FALSE | FALSE | 2022-06-27 |
| **png** | 0.1-7 | FALSE | FALSE | 2013-12-03 |
| **prettyunits** | 1.1.1 | FALSE | FALSE | 2020-01-24 |
| **processx** | 3.7.0 | FALSE | FALSE | 2022-07-07 |
| **profvis** | 0.3.7 | FALSE | FALSE | 2020-11-02 |
| **promises** | 1.2.0.1 | FALSE | FALSE | 2021-02-11 |
| **ps** | 1.7.1 | FALSE | FALSE | 2022-06-18 |
| **purrr** | 0.3.5 | FALSE | FALSE | 2022-10-06 |
| **R6** | 2.5.1 | FALSE | FALSE | 2021-08-19 |
| **rappdirs** | 0.3.3 | FALSE | FALSE | 2021-01-31 |
| **rbibutils** | 2.2.9 | FALSE | FALSE | 2022-08-15 |
| **Rcpp** | 1.0.9 | FALSE | FALSE | 2022-07-08 |
| **RCurl** | 1.98-1.9 | FALSE | FALSE | 2022-10-03 |
| **Rdpack** | 2.4 | FALSE | FALSE | 2022-07-20 |
| **remotes** | 2.4.2 | FALSE | FALSE | 2021-11-30 |
| **rlang** | 1.0.6 | FALSE | FALSE | 2022-09-24 |
| **rmarkdown** | 2.17 | FALSE | FALSE | 2022-10-07 |
| **RSQLite** | 2.2.18 | FALSE | FALSE | 2022-10-04 |
| **rstudioapi** | 0.14 | FALSE | FALSE | 2022-08-22 |
| **S4Vectors** | 0.35.4 | FALSE | FALSE | 2022-09-18 |
| **scales** | 1.2.1 | FALSE | FALSE | 2022-08-20 |
| **sessioninfo** | 1.2.2 | FALSE | FALSE | 2021-12-06 |
| **shiny** | 1.7.2 | FALSE | FALSE | 2022-07-19 |
| **stringi** | 1.7.8 | FALSE | FALSE | 2022-07-11 |
| **stringr** | 1.4.1 | FALSE | FALSE | 2022-08-20 |
| **synaptome.data** | 0.99.3 | TRUE | FALSE | 2022-08-22 |
| **synaptome.ldb** | 0.99.9 | TRUE | FALSE | 2022-10-17 |
| **tibble** | 3.1.8 | FALSE | FALSE | 2022-07-22 |
| **tidyselect** | 1.2.0 | FALSE | FALSE | 2022-10-10 |
| **urlchecker** | 1.0.1 | FALSE | FALSE | 2021-11-30 |
| **usethis** | 2.1.6 | FALSE | FALSE | 2022-05-25 |
| **utf8** | 1.2.2 | FALSE | FALSE | 2021-07-24 |
| **vctrs** | 0.4.2 | FALSE | FALSE | 2022-09-29 |
| **withr** | 2.5.0 | FALSE | FALSE | 2022-03-03 |
| **xfun** | 0.33 | FALSE | FALSE | 2022-09-12 |
| **xtable** | 1.8-4 | FALSE | FALSE | 2019-04-21 |
| **XVector** | 0.37.1 | FALSE | FALSE | 2022-08-25 |
| **yaml** | 2.3.5 | FALSE | FALSE | 2022-02-21 |
| **zlibbioc** | 1.43.0 | FALSE | FALSE | 2022-05-05 |

|  | source |
| --- | --- |
| **AnnotationDbi** | Bioconductor |
| **AnnotationHub** | Bioconductor |
| **assertthat** | CRAN (R 4.2.0) |
| **Biobase** | Bioconductor |
| **BiocFileCache** | Bioconductor |
| **BiocGenerics** | Bioconductor |
| **BiocManager** | CRAN (R 4.2.0) |
| **BiocVersion** | Bioconductor |
| **Biostrings** | Bioconductor |
| **bit** | CRAN (R 4.2.0) |
| **bit64** | CRAN (R 4.2.0) |
| **bitops** | CRAN (R 4.2.0) |
| **blob** | CRAN (R 4.2.0) |
| **cachem** | CRAN (R 4.2.0) |
| **callr** | CRAN (R 4.2.0) |
| **cli** | CRAN (R 4.2.0) |
| **colorspace** | CRAN (R 4.2.0) |
| **crayon** | CRAN (R 4.2.0) |
| **curl** | CRAN (R 4.2.1) |
| **DBI** | CRAN (R 4.2.0) |
| **dbplyr** | CRAN (R 4.2.0) |
| **devtools** | CRAN (R 4.2.1) |
| **digest** | CRAN (R 4.2.0) |
| **dplyr** | CRAN (R 4.2.1) |
| **ellipsis** | CRAN (R 4.2.0) |
| **evaluate** | CRAN (R 4.2.1) |
| **fansi** | CRAN (R 4.2.0) |
| **fastmap** | CRAN (R 4.2.0) |
| **filelock** | CRAN (R 4.2.0) |
| **fs** | CRAN (R 4.2.0) |
| **generics** | CRAN (R 4.2.0) |
| **GenomeInfoDb** | Bioconductor |
| **GenomeInfoDbData** | Bioconductor |
| **ggplot2** | CRAN (R 4.2.0) |
| **glue** | CRAN (R 4.2.0) |
| **gtable** | CRAN (R 4.2.1) |
| **htmltools** | CRAN (R 4.2.0) |
| **htmlwidgets** | CRAN (R 4.2.0) |
| **httpuv** | CRAN (R 4.2.0) |
| **httr** | CRAN (R 4.2.1) |
| **igraph** | CRAN (R 4.2.0) |
| **interactiveDisplayBase** | Bioconductor |
| **IRanges** | Bioconductor |
| **KEGGREST** | Bioconductor |
| **knitr** | CRAN (R 4.2.0) |
| **later** | CRAN (R 4.2.0) |
| **lifecycle** | CRAN (R 4.2.1) |
| **magrittr** | CRAN (R 4.2.0) |
| **memoise** | CRAN (R 4.2.0) |
| **mime** | CRAN (R 4.2.0) |
| **miniUI** | CRAN (R 4.2.0) |
| **munsell** | CRAN (R 4.2.0) |
| **pander** | CRAN (R 4.2.0) |
| **pillar** | CRAN (R 4.2.0) |
| **pkgbuild** | CRAN (R 4.2.0) |
| **pkgconfig** | CRAN (R 4.2.0) |
| **pkgload** | CRAN (R 4.2.0) |
| **png** | CRAN (R 4.2.0) |
| **prettyunits** | CRAN (R 4.2.0) |
| **processx** | CRAN (R 4.2.0) |
| **profvis** | CRAN (R 4.2.0) |
| **promises** | CRAN (R 4.2.0) |
| **ps** | CRAN (R 4.2.0) |
| **purrr** | CRAN (R 4.2.1) |
| **R6** | CRAN (R 4.2.0) |
| **rappdirs** | CRAN (R 4.2.0) |
| **rbibutils** | CRAN (R 4.2.0) |
| **Rcpp** | CRAN (R 4.2.0) |
| **RCurl** | CRAN (R 4.2.1) |
| **Rdpack** | CRAN (R 4.2.0) |
| **remotes** | CRAN (R 4.2.0) |
| **rlang** | CRAN (R 4.2.0) |
| **rmarkdown** | CRAN (R 4.2.1) |
| **RSQLite** | CRAN (R 4.2.0) |
| **rstudioapi** | CRAN (R 4.2.0) |
| **S4Vectors** | Bioconductor |
| **scales** | CRAN (R 4.2.0) |
| **sessioninfo** | CRAN (R 4.2.0) |
| **shiny** | CRAN (R 4.2.0) |
| **stringi** | CRAN (R 4.2.0) |
| **stringr** | CRAN (R 4.2.1) |
| **synaptome.data** | Bioconductor |
| **synaptome.ldb** | Bioconductor |
| **tibble** | CRAN (R 4.2.0) |
| **tidyselect** | CRAN (R 4.2.1) |
| **urlchecker** | CRAN (R 4.2.0) |
| **usethis** | CRAN (R 4.2.0) |
| **utf8** | CRAN (R 4.2.0) |
| **vctrs** | CRAN (R 4.2.0) |
| **withr** | CRAN (R 4.2.0) |
| **xfun** | CRAN (R 4.2.0) |
| **xtable** | CRAN (R 4.2.0) |
| **XVector** | Bioconductor |
| **yaml** | CRAN (R 4.2.0) |
| **zlibbioc** | Bioconductor |