keggorthology: the KEGG orthology as graph

VJ Carey

November 7, 2024

Contents

1 Introduction

KEGG is the Kyoto Encyclopedia of Genes and Genomes. An important product of the KEGG group is a catalog of pathways. The KEGG Orthology (KO) organizes the pathways into a conceptual hierarchy. This package encodes the hierarchy as a graph, and provides some support for deriving sets of array feature identifiers from the hierarchy.

2 KOgraph

```
> library(keggorthology)
> library(graph)
> data(KOgraph)
> KOgraph

A graphNEL graph with directed edges
Number of Nodes = 358
Number of Edges = 357

> nodes(KOgraph)[1:5]
```

```
[1] "KO.Feb10root"                 "Metabolism"
[3] "Carbohydrate Metabolism"      "Glycolysis / Gluconeogenesis"
[5] "Citrate cycle (TCA cycle)"
```

The upper component of the hierarchy is:

```
> adj(KOgraph, nodes(KOgraph)[1])

$KO.Feb10root
[1] "Metabolism"
[2] "Genetic Information Processing"
[3] "Environmental Information Processing"
[4] "Cellular Processes"
[5] "Organismal Systems"
[6] "Human Diseases"
```

Graph operations can be used to explore the orthology. For example, the context of the PPAR signaling pathway is found as follows:

```
> library(RBGL)
> sp.between(KOgraph, nodes(KOgraph)[1], "PPAR signaling pathway")

$`KO.Feb10root:PPAR signaling pathway`
$`KO.Feb10root:PPAR signaling pathway`$length
[1] 3

$`KO.Feb10root:PPAR signaling pathway`$path_detail
[1] "KO.Feb10root"           "Organismal Systems"     "Endocrine System"
[4] "PPAR signaling pathway"

$`KO.Feb10root:PPAR signaling pathway`$length_detail
$`KO.Feb10root:PPAR signaling pathway`$length_detail[[1]]
        KO.Feb10root->Organismal Systems
                                       1
    Organismal Systems->Endocrine System
                                       1
Endocrine System->PPAR signaling pathway
                                       1
```

Fixed-length identifiers are used to label pathways. These are available as the 'tag' nodeData attribute.

```
> nodeData(KOgraph,,"tag")[1:5]
```

```
$KO.Feb10root
[1] "NONE"

$Metabolism
[1] "01100"

$`Carbohydrate Metabolism`
[1] "01101"

$`Glycolysis / Gluconeogenesis`
[1] "00010"

$`Citrate cycle (TCA cycle)`
[1] "00020"
```

The depth of each term is also available.

```
> nodeData(KOgraph,,"depth")[1:5]
```

```
$KO.Feb10root
[1] 0

$Metabolism
[1] 1

$`Carbohydrate Metabolism`
[1] 2

$`Glycolysis / Gluconeogenesis`
[1] 3

$`Citrate cycle (TCA cycle)`
[1] 3
```

3 Application to gene filtering

Several functions are available for retrieving relevant information from the orthology. If you know a substring of the pathway name of interest, you can obtain the numerical tag(s).

```
> getKOtags("insulin")
```

```
Insulin signaling pathway
                  "04910"
```

We can get probe set identifiers corresponding to a term. The default chip annotation package used is hgu95av2.db.

```
> library(hgu95av2.db)
> mp = getKOprobes("Methionine")
> library(ALL)
> data(ALL)
> ALL[mp,]

ExpressionSet (storageMode: lockedEnvironment)
assayData: 30 features, 128 samples
  element names: exprs
protocolData: none
phenoData
  sampleNames: 01005 01010 ... LAL4 (128 total)
  varLabels: cod diagnosis ... date last seen (21 total)
  varMetadata: labelDescription
featureData: none
experimentData: use 'experimentData(object)'
  pubMedIds: 14684422 16243790
Annotation: hgu95av2
```

4 Infrastructure considerations

Based on keggorthology read of KEGG orthology, March 2 2010. Specifically, we run wget on `ftp://ftp.genome.jp/pub/kegg/brite/ko/ko00001.keg` and use parsing and modeling code given in inst/keggHTML to generate a data frame respecting the hierarchy, and then keggDF2graph function in keggorthology package to construct the graph.

5 Session info

```
> sessionInfo()

R version 4.4.2 (2024-10-31)
Platform: x86_64-pc-linux-gnu
Running under: Ubuntu 24.04.1 LTS

Matrix products: default
BLAS:   /usr/lib/x86_64-linux-gnu/openblas-pthread/libblas.so.3
LAPACK: /usr/lib/x86_64-linux-gnu/openblas-pthread/libopenblasp-r0.3.26.so;  LAPACK ver

locale:
 [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
 [3] LC_TIME=en_US.UTF-8        LC_COLLATE=C
 [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
 [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
 [9] LC_ADDRESS=C               LC_TELEPHONE=C
[11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C

time zone: Etc/UTC
tzcode source: system (glibc)

attached base packages:
[1] stats4    stats     graphics  grDevices utils     datasets  methods
[8] base

other attached packages:
 [1] ALL_1.48.0           RBGL_1.83.0          keggorthology_2.59.0
 [4] graph_1.85.0         hgu95av2.db_3.13.0   org.Hs.eg.db_3.20.0
 [7] AnnotationDbi_1.69.0 IRanges_2.41.0       S4Vectors_0.45.0
[10] Biobase_2.67.0       BiocGenerics_0.53.1  generics_0.1.3

loaded via a namespace (and not attached):
 [1] bit_4.5.0               jsonlite_1.8.9          compiler_4.4.2
 [4] crayon_1.5.3            blob_1.2.4              Biostrings_2.75.1
 [7] png_0.1-8               fastmap_1.2.0           R6_2.5.1
[10] XVector_0.47.0          GenomeInfoDb_1.43.0     knitr_1.48
[13] maketools_1.3.1         GenomeInfoDbData_1.2.13 DBI_1.2.3
[16] rlang_1.1.4             KEGGREST_1.47.0         cachem_1.1.0
[19] xfun_0.49               sys_3.4.3               bit64_4.5.2
[22] RSQLite_2.3.7           memoise_2.0.1           cli_3.6.3
[25] zlibbioc_1.52.0         vctrs_0.6.5             buildtools_1.0.0
[28] httr_1.4.7              tools_4.4.2             pkgconfig_2.0.3
[31] UCSC.utils_1.3.0
```