

# *keggorthology*: the KEGG orthology as graph

VJ Carey

April 30, 2024

## Contents

|          |                                      |          |
|----------|--------------------------------------|----------|
| <b>1</b> | <b>Introduction</b>                  | <b>1</b> |
| <b>2</b> | <b>KOgraph</b>                       | <b>1</b> |
| <b>3</b> | <b>Application to gene filtering</b> | <b>3</b> |
| <b>4</b> | <b>Infrastructure considerations</b> | <b>4</b> |
| <b>5</b> | <b>Session info</b>                  | <b>4</b> |

## 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 = getK0probes("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 4.4.0 beta (2024-04-14 r86421)
Platform: x86_64-apple-darwin20
Running under: macOS Monterey 12.7.1
```

```
Matrix products: default
```

```
BLAS: /Library/Frameworks/R.framework/Versions/4.4-x86_64/Resources/lib/libRblas.0.dylib
LAPACK: /Library/Frameworks/R.framework/Versions/4.4-x86_64/Resources/lib/libRlapack.dylib
```

```
locale:
```

```
[1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
```

```
time zone: America/New_York
```

```
tzcode source: internal
```

```
attached base packages:
```

```
[1] stats4      stats      graphics  grDevices  utils      datasets  methods
```

```
[8] base
```

```
other attached packages:
```

```
[1] ALL_1.45.0          RBGL_1.80.0          keggorthology_2.56.0
```

```
[4] hgu95av2.db_3.13.0  org.Hs.eg.db_3.19.1  AnnotationDbi_1.66.0
```

```
[7] IRanges_2.38.0      S4Vectors_0.42.0     Biobase_2.64.0
```

```
[10] graph_1.82.0        BiocGenerics_0.50.0
```

```
loaded via a namespace (and not attached):
```

```
[1] crayon_1.5.2          vctrs_0.6.5          httr_1.4.7
```

```
[4] cli_3.6.2             rlang_1.1.3          DBI_1.2.2
```

```
[7] png_0.1-8            UCSC.utils_1.0.0     jsonlite_1.8.8
```

```
[10] bit_4.0.5            Biostrings_2.72.0    KEGGREST_1.44.0
```

```
[13] fastmap_1.1.1         GenomeInfoDb_1.40.0  memoise_2.0.1
```

```
[16] compiler_4.4.0       RSQLite_2.3.6        blob_1.2.4
```

```
[19] pkgconfig_2.0.3      XVector_0.44.0       R6_2.5.1
```

```
[22] GenomeInfoDbData_1.2.12 tools_4.4.0          bit64_4.0.5
```

```
[25] zlibbioc_1.50.0      cachem_1.0.8
```