

KEGG Tutorial

Ajay Penugonda

June 2024

1 Introduction

This tutorial will guide you through accessing the Kyoto Encyclopedia of Genes and Genomes (KEGG) using the **BioServices** Python package. KEGG is a comprehensive database resource for understanding high-level functions and utilities of the biological system. **BioServices** is a Python package that provides a programmatic interface to many bioinformatics databases.

2 Installation

First, you need to install the **bioservices** package. You can install it using `pip`. This package will allow you to interact with KEGG programmatically.

```
!pip install bioservices
```

3 Importing the KEGG Module

After installation, you can import the KEGG service from the **bioservices** package and create an instance of the KEGG service.

```
from bioservices import KEGG
```

```
k = KEGG()
```

4 Basic Usage

4.1 List Available Databases

To get a list of databases available in KEGG, use the `list_databases` method. This provides an overview of the different databases that you can access within KEGG.

```
databases = k.list_databases()  
print(databases)
```

4.2 Search for Entries

To search for specific entries within the KEGG databases, you can use methods such as `list` and `find`. For example, to find all entries related to a specific query, you could do:

```
entries = k.find("compound", "glucose")
print(entries)
```

4.3 Get Entry Information

To retrieve detailed information about a specific entry, use the `get` method. For example, to get information on the KEGG pathway for insulin signaling (hsa04910):

```
entry = k.get("pathway:hsa04910")
print(entry)
```

This returns detailed information about the specified pathway, including its components and their interactions.

4.4 Get a List of Pathways for an Organism

To get a list of pathways for a specific organism, use the `list` method with the organism code (e.g., 'hsa' for Homo sapiens):

```
pathways = k.list("pathway", "hsa")
print(pathways)
```

This provides a comprehensive list of all pathways associated with the specified organism.

5 Advanced Usage

5.1 Parse KEGG Pathway

The information retrieved from KEGG can be in various formats. For example, to parse the pathway map for visualization or further analysis, you can use the `parse_kgml_pathway` method:

```
from bioservices.kegg import KEGGParser

pathway_data = k.get("pathway:hsa04910", "kgml")
parser = KEGGParser()
pathway = parser.parse_kgml_pathway(pathway_data)
print(pathway)
```

This allows you to convert the KGML (KEGG Markup Language) data into a more usable form for analysis.

5.2 Visualize Pathways

You can also visualize pathways using additional tools. For instance, you could use **networkx** for network visualization or other specialized bioinformatics visualization libraries. Here is an example of how you might start visualizing a pathway:

```
import networkx as nx
import matplotlib.pyplot as plt

G = nx.DiGraph()

# Add nodes and edges from the parsed pathway data
for node in pathway.entries.values():
    G.add_node(node.name, type=node.type)

for rel in pathway.relations:
    G.add_edge(rel.entry1.name, rel.entry2.name, type=rel.type)

# Draw the network
nx.draw(G, with_labels=True)
plt.show()
```

This creates a directed graph of the pathway using **networkx**.

6 Conclusion

This tutorial provides an in-depth guide to accessing and utilizing the KEGG database with the **BioServices** Python package. By following the steps outlined, you can programmatically interact with KEGG to retrieve, parse, and visualize biological pathway data.

For further reading and exploration, consider delving into the **BioServices** documentation and exploring other bioinformatics tools that can complement your data analysis workflow.