Project Design

Genome 559: Introduction to Statistical and Computational Genomics

Elhanan Borenstein
Hypothesis:
The average degree in the metabolic networks of Prokaryotes is higher than the average degree in the metabolic networks of Eukaryotes
KEGG FTP

KEGG FTP Site for Academic Users

The KEGG data may be downloaded by academic users from the KEGG FTP site:


Non-academic users are required to obtain a license agreement for downloading KEGG.

- Terms of use
- Licensing from Pathway Solutions

**Announcement:**
A new directory, "module", is created.

Posted on December 22, 2010  » Past announcements

**Directories and Files**

- **pathway/**
  - KEGG PATHWAY (daily updated)
- **map/**
  - Reference pathway maps
- **ko/**
  - Reference pathway maps (KO)
- **ec/**
  - Reference pathway maps (EC)
- **rn/**
  - Reference pathway maps (reaction)
- **organisms/**
  - Organism-specific pathway maps
- **pathway**
  - Pathway entries (text data)
- **map_title.tab**
  - List of pathways available

- **module/**
  - KEGG MODULE (daily updated) New!
  - Reference module maps (KO) - to be added
  - Organism-specific module maps - to be added
  - Module entries (text data)
<table>
<thead>
<tr>
<th>ENTRY</th>
<th>K00001</th>
<th>K00002</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>E1.1.1.1, adh</td>
<td>E1.1.1.2, adh</td>
</tr>
<tr>
<td>DEFINITION</td>
<td>alcohol dehydrogenase [EC:1.1.1.1]</td>
<td>alcohol dehydrogenase (NADP+) [EC:1.1.1.2]</td>
</tr>
<tr>
<td>PATHWAY</td>
<td>ko00010 Glycolysis / Gluconeogenesis ko00071 Fatty acid metabolism</td>
<td>ko00010 Glycolysis / Gluconeogenesis ko00561 Glycerolipid metabolism</td>
</tr>
<tr>
<td>MODULE</td>
<td>M00236 Retinol biosynthesis, beta-cacrotene =&gt; retinol</td>
<td></td>
</tr>
<tr>
<td>CLASS</td>
<td>Metabolism; Carbohydrate Metabolism; Glycolysis / Gluconeogenesis</td>
<td>Metabolism; Carbohydrate Metabolism; Glycolysis / Gluconeogenesis</td>
</tr>
<tr>
<td>DBLINKS</td>
<td>RN: R00623 R00754 R02124 R04880 R05233 R05234 R06917 R06927 R07105 R08281 R08306 R08310</td>
<td></td>
</tr>
<tr>
<td></td>
<td>COG: COG1012 COG1062 COG1064 COG1454</td>
<td></td>
</tr>
<tr>
<td></td>
<td>GO: 0004022 0004023 0004024 0004025</td>
<td></td>
</tr>
<tr>
<td>GENES</td>
<td>HSA: 124 (ADH1A) 125 (ADH1B) 126 (ADH1C) 127 (ADH4) 130 (ADH6) 131 (ADH7)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PTR: 461394 (ADH4) 461395 (ADH6) 461396 (ADH1B) 471257 (ADH7) 744064 (ADH1A) 744176 (ADH1C)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MCC: 707367 707682 (ADH1A) 708520 711061 (ADH1C)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PAS: Pars_0396 Pars_0534 Pars_0547 Pars_1545 Pars_2114</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TPE: Tpen_1006 Tpen_1516</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td>-----</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0001</td>
<td>ko:K02313</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0002</td>
<td>ko:K02338</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0003</td>
<td>ko:K03629</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0005</td>
<td>ko:K02470</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0006</td>
<td>ko:K02469</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0012</td>
<td>ko:K03767</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0018</td>
<td>ko:K01664</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0019</td>
<td>ko:K08884</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0020</td>
<td>ko:K05364</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0026</td>
<td>ko:K01552</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0029</td>
<td>ko:K00111</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0031</td>
<td>ko:K00627</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0032</td>
<td>ko:K00162</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0033</td>
<td>ko:K00161</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0035</td>
<td>ko:K00817</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0036</td>
<td>ko:K07448</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0039</td>
<td>ko:K04750</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0041</td>
<td>ko:K03281</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0048</td>
<td>ko:K08323</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0051</td>
<td>ko:K03734</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0052</td>
<td>ko:K03147</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0057</td>
<td>ko:K03088</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0059</td>
<td>ko:K01010</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0061</td>
<td>ko:K03711</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0062</td>
<td>ko:K06980</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0063</td>
<td>ko:K07560</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0072</td>
<td>ko:K12373</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0075</td>
<td>ko:K01834</td>
<td></td>
</tr>
<tr>
<td>ace:Ace1_0076</td>
<td>ko:K09796</td>
<td></td>
</tr>
</tbody>
</table>

...
Designing with Pseudo-Code Comments
Top down approach

# Preprocessing
# =============

# Build networks and calc degree
# ============

# Print output
# =============
# Preprocessing
# =============

# Read and store mapping from KO to RN

# Read and store mapping from RN to edges

# Build networks and calc degree
# =============

# Loop over species

# Read KO list of current species

# Map KO to RN and RN to edges

# Calculate degree

# Store: species, degree, phyla

# Print output
# =============

# Read and store species list and lineages

# Calculated average degree per P and per E

# Print
Add notes to self

# Preprocessing
# =============

# Read and store mapping from KO to RN

# Read and store mapping from RN to edges

# Read and store species list and lineages

# Build networks and calc degree
# ==============================

# Loop over species

# Read KO list of current species

# Map KO to RN and RN to edges

# -> Here I should have a full network
# -> TBD: What data structure should I use?

# Calculate degree

# -> TBD: How do I store results?

# Store: species, degree, phyla
# -> TBD: How do I store results?

# Print output
# ============

# Calculated average degree per P and per E

# Print
Add variables, loops, if-s, function calls

```python
# Preprocessing
# =============

# Read and store mapping from KO to RN
KO_file = 'ko.txt'
KO_to_RN = {}

# Read and store mapping from RN to edges
RN_file = 'reaction.txt'
RN_to_EDGES = {}

# Read and store species list and lineages
Genomes_file = 'genome.txt'
species_list = []
species_lineage = {}

# Build networks and calc degree
# =============

# Loop over species
for species in species_list:
    # Read KO list of current species
    # Map KO to RN and RN to edges
    # Here I should have a full network
    # TBD: What data structure should I use?
    # Calculate degree
    degree = CalcDegree(network)

    # Store: species, degree, phyla
    # TBD: How do I store results?

# Print output
# =============

# Calculated average degree per P and per E

# Print
```
Start coding small chunks

# Preprocessing
# =============

# Read and store mapping from KO to RN
KO_file = 'ko.txt'
KO_to_RN = {}

# Read and store mapping from RN to edges
RN_file = 'reaction.txt'
RN_to_EDGES = {}

# Read and store species list and lineages
Genomes_file = 'genome.txt'
species_list = []
species_lineage = {}
Define interfaces

# Preprocessing
# =============

# Read and store mapping from KO to RN
KO_file = 'ko.txt'
KO_to_RN = {}

# Read and store mapping from RN to edges
RN_file = 'reaction.txt'
RN_to_EDGES = {}

# Read and store species list and lineages
Genomes_file = 'genome.txt'
species_list = []
species_lineage = {}< LET’S WRITE THIS PART >

# Build networks and calc degree
# ===============

# Loop over species
for species in species_list:
  # Read KO list of current species
  # Map KO to RN and RN to edges
  # Here I should have a full network
  # TBD: What data structure should I use?
  # Calculate degree
  degree = CalcDegree(network)
  # Store: species, degree, phyla
  # TBD: How do I store results?

# Print output
# =============

# Calculated average degree per P and per E

# Print
Computational Representation of Networks

List of edges: (ordered) pairs of nodes

\[
\begin{array}{l}
(A, C), (C, B), (D, B), (D, C)
\end{array}
\]

Connectivity Matrix

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>B</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>D</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Object Oriented

- Name: A
  - ngr: [p1]
- Name: B
  - ngr: [p1, p2]
- Name: C
  - ngr: [p1]
- Name: D
  - ngr: [p1]

Which is the most useful representation?
... it’s a wrap ...
Hope you enjoyed!