Python for Life Sciences

Kulwadee Somboonviwat

kulwadee.a [at] gmail.com

PyCon Thailand 2018-06-16

Slide available at: https://goo.gl/6fqDAL

Me

Current

- Software developer working on bioinformatics and dialog systems

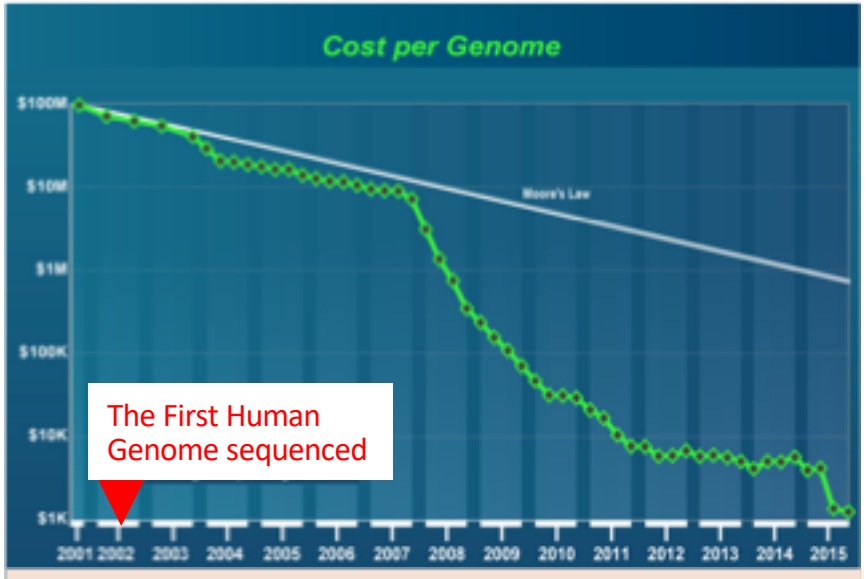
Past

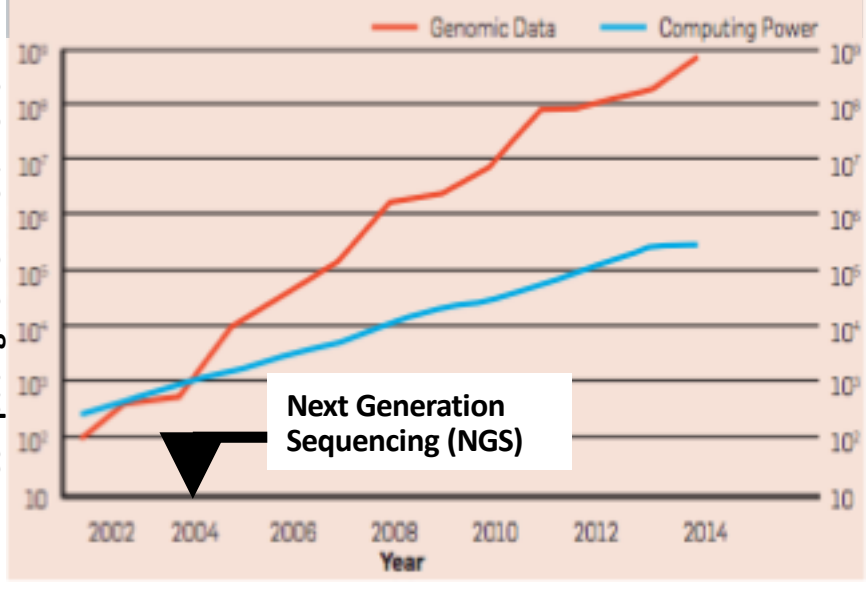
- Faculty member at KMUTNB, KMITL, KMUTT
- Specially Appointed Assoc.Prof. at University of Electro-Communications (UEC)
- Technical consultant (PEA)
- Part-time researcher at CoE for Molecular Biology and Genomic of Shrimp (CU)
- Ph.D. in Information and Communication Engineering (Univ. of Tokyo)

Today's Topics

- Why Python for Life Sciences?
- Important Concepts
  - The Central Dogma of Molecular Biology
  - Types of Biological Data
- Common Tasks in Bioinformatics
  - Python Libraries and Tools
  - Example: Gene Expression Analysis
- Key Issues in doing Biological Data Science (a computer scientist perspective)

Life Sciences is a Big Data Science.





Explosion of Genomic Data

With the advent of **NGS** in 2004…

- Cost-per-Genome → **US$1000**
- **10x** yearly growth rate of Genomic Data

https://www.genome.gov/27565109/the-cost-of-sequencing-a-human-genome/

B., Bonnie, N. M. Daniels, and Y. W. Yu. "Computational Biology in the 21st Century: Scaling with Compressive Algorithms." *CACM* 59.8: 72-80.

Big Data: Astronomical or Genomical?

Zachary D. Stephens, Skylar Y. Lee, Faraz Faghri, Roy H. Campbell, Chengxiang Zhai, Miles J. Efron, Ravishankar Iyer, Michael C. Schatz, Saurabh Sinha, Gene E. Robinson

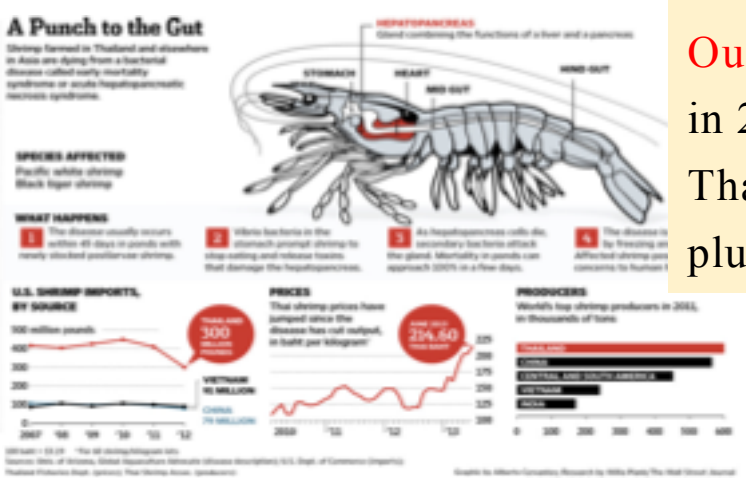
Published: July 7, 2015 • https://doi.org/10.1371/journal.pbio.1002195

"**Genomics** is either on par with or the most demanding of the domains analyzed (**Astronomy**, **Twitter**, **YouTube**) in terms of data acquisition, storage, distribution, and analysis. "

Data Phase	Astronomy	Twitter	YouTube	Genomics
Acquisition	25 zetta-bytes/year	0.5–15 billion tweets/year	500–900 million hours/year	1 zetta-bases/year
Storage	1 EB/year	1–17 PB/year	1–2 EB/year	2–40 EB/year
Analysis	In situ data reduction	Topic and sentiment mining	Limited requirements	Heterogeneous data and analysis
	Real-time processing	Metadata analysis		Variant calling, ~2 trillion central processing unit (CPU) hours
	Massive volumes			All-pairs genome alignments, ~10,000 trillion CPU hours
Distribution	Dedicated lines from antennae to server (600 TB/s)	Small units of distribution	Major component of modern user's bandwidth (10 MB/s)	Many small (10 MB/s) and fewer massive (10 TB/s) data movement

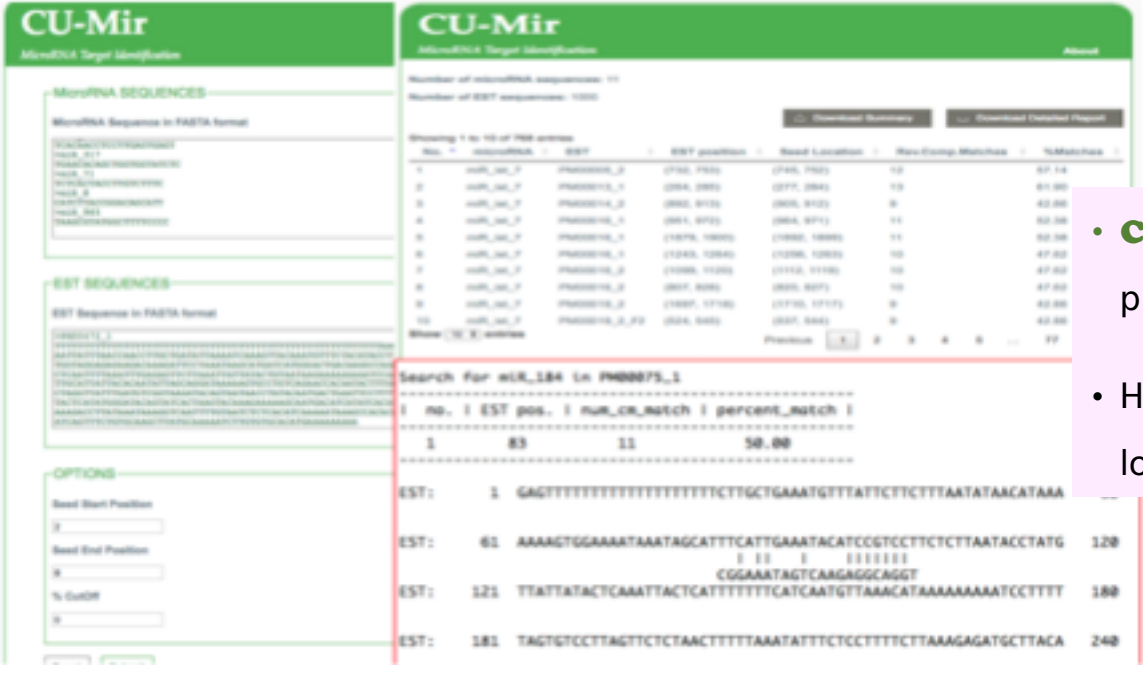
Biological Data Science has **huge impact** on society.

https://goo.gl/BCAcfH



Outbreak of shrimp diseases in 2012 to 2015 had caused Thai shrimp production plummeted as much as 40%

https://goo.gl/3PytXP



- **CU-Mir** is a miRNA target identification program, written in Python.
- Help scientists to be more productive by lowering time and costs of experiments

https://www.ncbi.nlm.nih.gov/pubmed/26945623

Other emerging applications ...

Business Impact

Oxford Nanopore's Hand-Held DNA Analyzer Has Traveled the World

https://nanoporetech.com/applications

A British company, number 32 on our list of the 50 Smartest Companies, bets a tiny analyzer will change how we look at DNA.

by Antonio Regalado June 27, 2017

https://goo.gl/oisL5o

Rewriting Life

This new company wants to sequence your genome and let you share it on a blockchain

People will be able to earn cryptocurrency in exchange for letting pharma companies use their data.

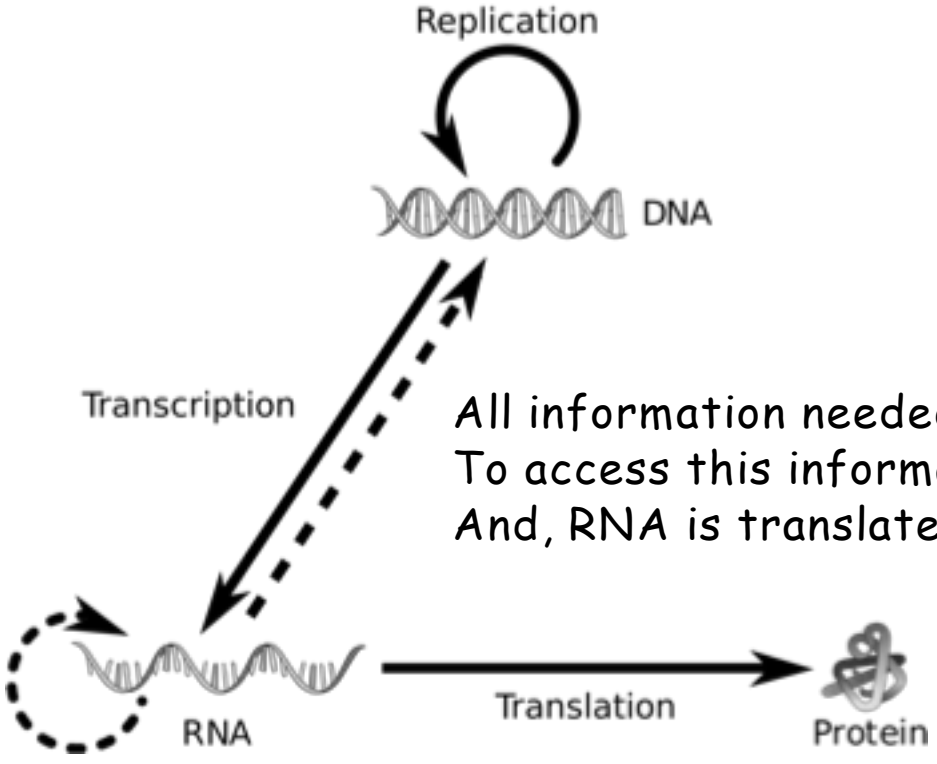
by Emily Mullin February 7, 2018

https://goo.gl/qnVMuZ

Today's Topics

- Why Python for Life Sciences?
- Important Concepts
  - The Central Dogma of Molecular Biology
  - Types of Biological Data
- Common Tasks in Bioinformatics
  - Python Libraries and Tools
  - Example: Gene Expression Analysis
- Key Issues in doing Biological Data Science (a computer scientist perspective)

The Central Dogma of Molecular Biology



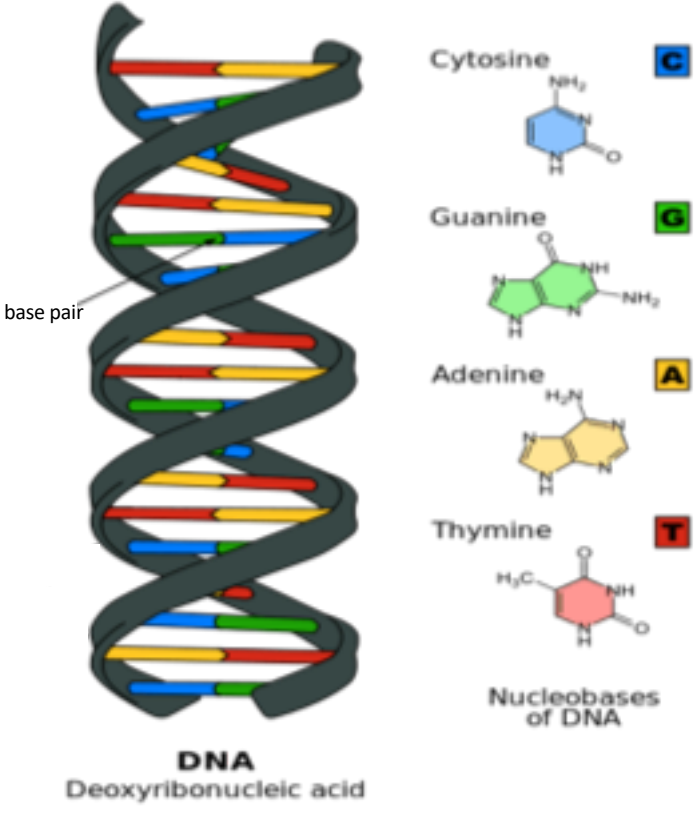
All information needed to run a cell is kept in **DNA**.
To access this information, the DNA is transcribed into **RNA**.
And, RNA is translated into **protein**

Video animation by RIKEN: https://www.youtube.com/watch?v=J3HVVi2k2No

Types of Biological Data

- DNA, RNA, Protein Sequences
- Gene Expression
- Networks and Pathways
- Gene Ontology

Types Biological Data – DNA sequence

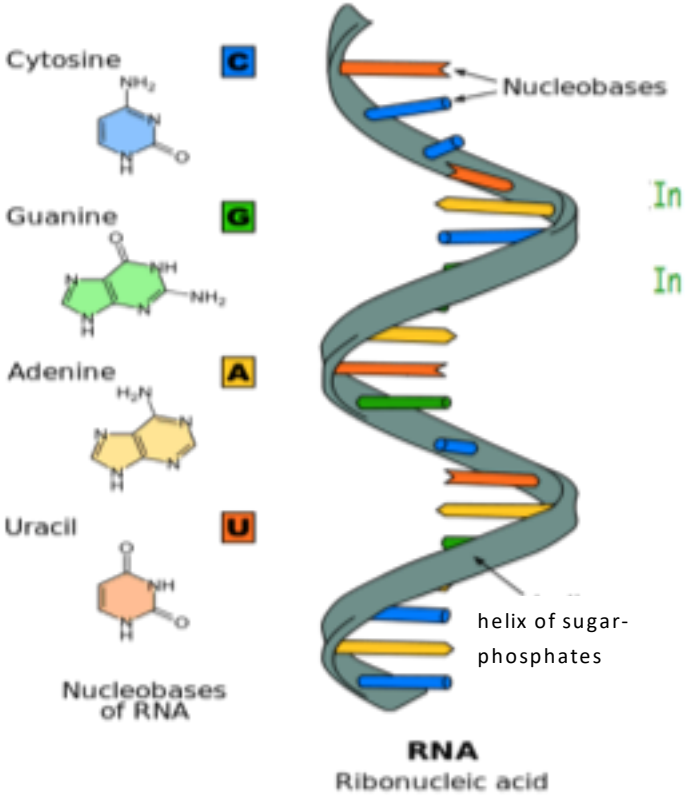



```
In [28]: dnaAlphabet = ['a','c','t','g']

In [29]: dnaSequence = "caaccagcccaaggccatccgaggttctcacagttactagaggatgggtgccagg
    ...: cagttctcaggggtatacgcctgttggtcaggcggcgatgtatttctgtccgcgtcgcgtgagtgcccac
    ...: gcgaatcgaaatgcagattatgagcttttccgagacacacctctgcagaatcggccgcacgggcgcgcgt
    ...: tattgcctccttagggaattctaattccacctcaccggcatacgtcggagcggagttatggacctctttg
    ...: taattgcaagggatctgaggtcagacaagcagcat"
```

https://www.thoughtco.com/dna-vs-rna-1224518

Types Biological Data – RNA sequence




```
In [40]: rnaAlphabet = ['a','c','u','g']

In [41]: rnaSequence = "caaccagcccaaggccauccgaggguucucacaguuacuagaggaugggugccagg
    ...: caguucucaggguauacgccuguuggucaggcggcgauguauuucuguccgcgucgcgugagugcccac
    ...: gcgaaucgaaaugcagauuaugagcuuuuccgagacacaccucugcagaaucggccgcacgggcgcgcgu
    ...: uauugccuccuuagggaauucuaauuccaccucaccggcauacgucggagcggaguuauggaccucuuug
    ...: uaauugcaagggaucugaggucagacaagcagcau"
```

https://www.thoughtco.com/dna-vs-rna-1224518

Types Biological Data – Protein

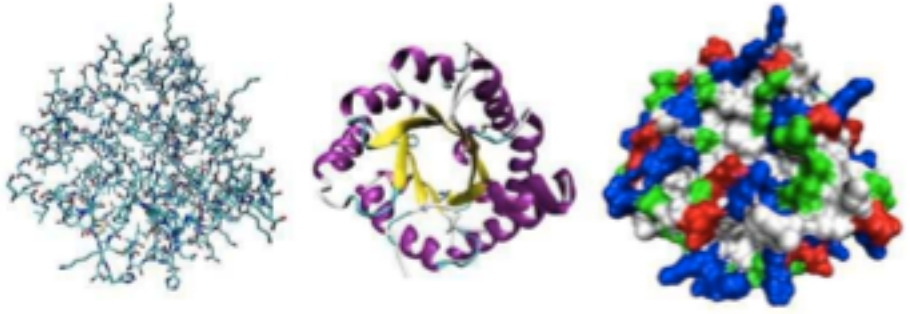
RNA codon table

1st position	2nd position U	2nd position C	2nd position A	2nd position G	3rd position
U	Phe	Ser	Tyr	Cys	U
	Phe	Ser	Tyr	Cys	C
	Leu	Ser	stop	stop	A
	Leu	Ser	stop	Trp	G
C	Leu	Pro	His	Arg	U
	Leu	Pro	His	Arg	C
	Leu	Pro	Gln	Arg	A
	Leu	Pro	Gln	Arg	G
A	Ile	Thr	Asn	Ser	U
	Ile	Thr	Asn	Ser	C
	Ile	Thr	Lys	Arg	A
	Met	Thr	Lys	Arg	G
G	Val	Ala	Asp	Gly	U
	Val	Ala	Asp	Gly	C
	Val	Ala	Glu	Gly	A
	Val	Ala	Glu	Gly	G

Amino Acids

Ala: Alanine
Arg: Arginine
Asn: Asparagine
Asp: Aspartic acid
Cys: Cysteine
Gln: Glutamine
Glu: Glutamic acid
Gly: Glycine
His: Histidine
Ile: Isoleucine
Leu: Leucine
Lys: Lysine
Met: Methionine
Phe: Phenylalanine
Pro: Proline
Ser: Serine
Thr: Threonine
Trp: Tryptophane
Tyr: Tyrosine
Val: Valine

http://knowgenetics.org/genetic-code/

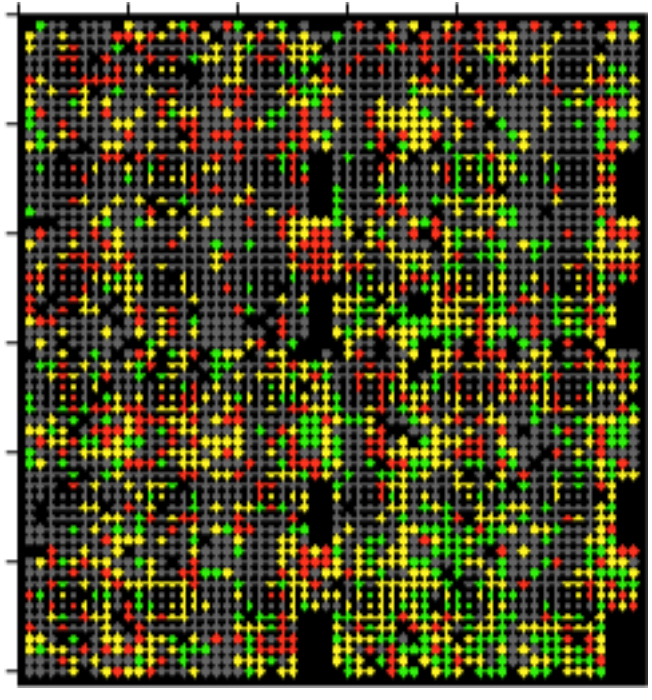


```
In [48]: proteinAlphabet = ['R','K','D','E','Q','N','H','S',
    ...: 'T','Y','C','W','A','I','L','M','F','V','P','G']

In [49]: proteinSequence = "KHHKYTGLFQSRSAQFWGAFKRLHLSENRARS
    ...: SNTGQDARAESWNKIVHVTADASDQDKQAEMEPSNLSYEHVHPRHQYPWVE
    ...: KKMQIECQTDLGHNVEHQRGTLSRCMQDKPRKIDATGSFDGPPNCGSQFVA
    ...: MRKCSWSTPPSPGDLQSDCPVTENHLGTDMWVDTNKEDAMPVMWGSDMVWC
    ...: DSGQNHKHYPWKKFFTWCRYGPACDMCYHMTMQILTDLNAVMLIVAWTPRD
    ...: WAMSNLWFSLPRGPEVEAHTLIWMLMKVHHVKDWWLRSNGFAMMGWNEGPW
    ...: PTSECFTINLCRYHMLDVCQCWCNITMTRRVTWIESDPKNKEFPAANFYYE
    ...: RQNEGQYGAIHVIRGPYLQAGTSEENFHTFQQYSSTGPEKVPIDHTCMCDV
    ...: SEQHCQEAIGVGKRQVKYNHAEQYLRCCWEIDELVTLYTHMFSEQRMHILP
    ...: DIVDFMYRGMFVCREFNHDMRPAQDRWNKKMQAFGARNEG"
```

Types Biological Data – Gene Expression

- **Gene.** A stretch of DNA that codes for a type of protein that has a function in an organism
- **Gene Expression Data.** Expression level of genes in an individual that is measured through *Microarray*



(n x m) Data Matrix

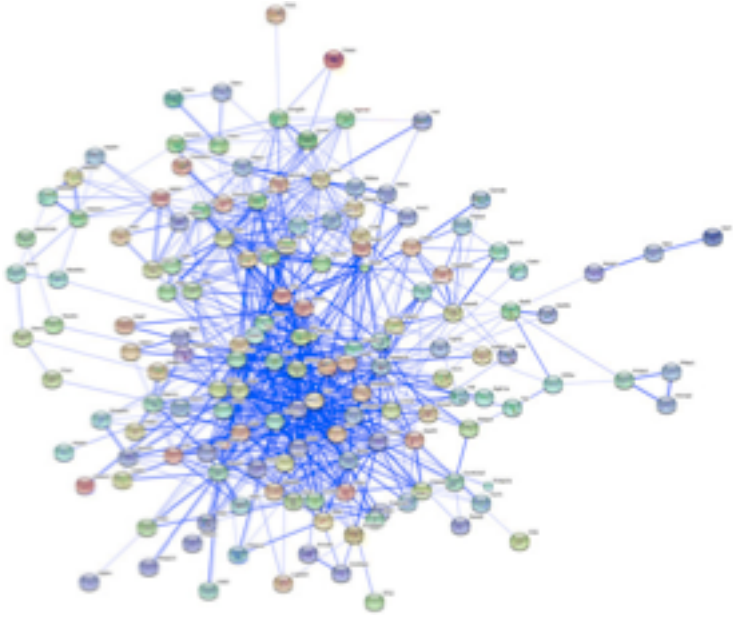
m Samples

Gene	Sample 1	Sample 1	.....	Sample m
a				
b				
c				
...				
n				

n Samples

Types Biological Data – Networks and Pathways

Co-Expression Network



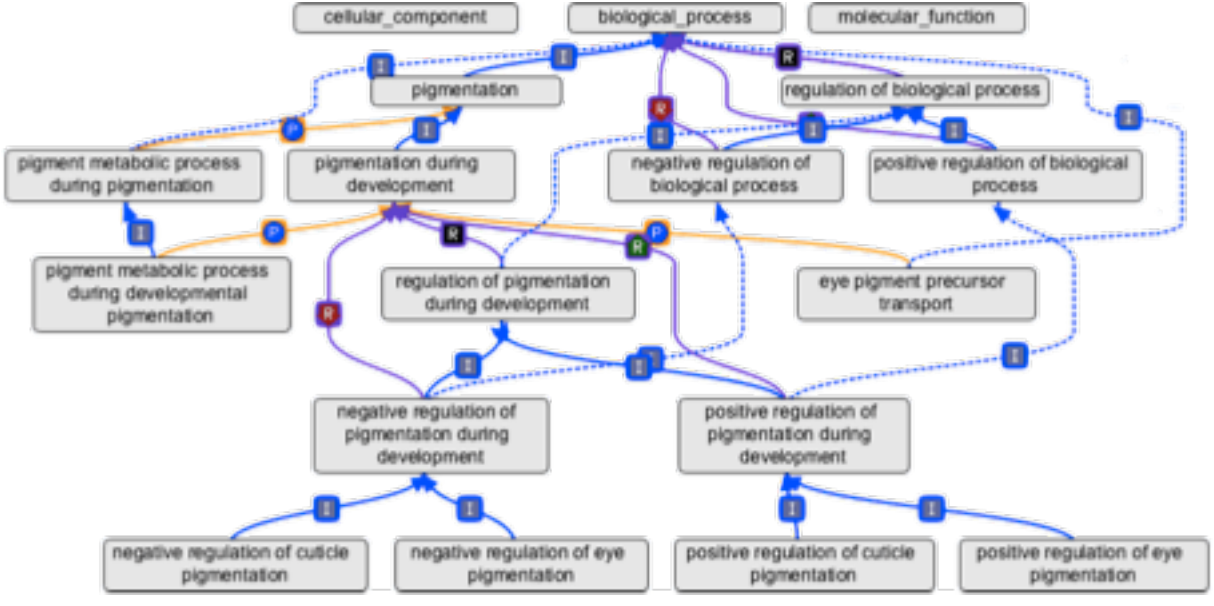
Protein-Protein Interaction Network

For more details, pls. refer to https://goo.gl/mnAL1f

Types Biological Data – Gene Ontology

Gene Ontology defines the universe of concepts relating to gene functions (‘GO terms’), and how these functions are related to each other (‘relations’).

http://geneontology.org/page/ontology-documentation



Today's Topics

- Why Python for Life Sciences?
- Important Concepts
  - The Central Dogma of Molecular Biology
  - Types of Biological Data
- Common Tasks in Bioinformatics
  - Python Libraries and Tools
  - Example: Gene Expression Analysis
- Key Issues in doing Biological Data Science (a computer scientist perspective)

Python Libraries and Tools

- Jupyter Notebook
- scipy, numpy, matplotlib, pandas
- scikit-learn, scikit-image
- rpy2
- biopython, scikit-bio

Differential Expression Analysis with Python and Bioconductor

Credits: https://goo.gl/XPV6Ji

- Use **numpy** to store expression data
- Interface with Bioconductor (an R package) via **rpy2**

Read Expression Data and Store it in numpy array

```
import csv
import collections

def read_count_file(in_file):
    """Read count information from a simple CSV file into a dictionary.
    """
    counts = collections.defaultdict(dict)
    with open(in_file) as in_handle:
        reader = csv.reader(in_handle)
        header = reader.next()
        conditions = header[1:]
        for parts in reader:
            region_name = parts[0]
            region_counts = [float(x) for x in parts[1:]]
            for ci, condition in enumerate(conditions):
                counts[condition][region_name] = region_counts[ci]
    return dict(counts)
```

```
import numpy

def get_conditions_and_genes(work_counts):
    conditions = work_counts.keys()
    conditions.sort()
    all_genes = []
    for c in conditions:
        all_genes.extend(work_counts[c].keys())
    all_genes = list(set(all_genes))
    all_genes.sort()
    sizes = [work_counts[c]["Total"] for c in conditions]
    all_genes.remove("Total")
    return conditions, all_genes, sizes

def edger_matrices(work_counts):
    conditions, all_genes, sizes = get_conditions_and_genes(work_counts)
    assert len(sizes) == 2
    groups = [1, 2]
    data = []
    final_genes = []
    for g in all_genes:
        cur_row = [int(work_counts[c][g]) for c in conditions]
        if sum(cur_row) > 0:
            data.append(cur_row)
            final_genes.append(g)
    return (numpy.array(data), numpy.array(groups), numpy.array(sizes),
            conditions, final_genes)
```

Use rpy2 to access a DGEList and call topTags function in R

```
import rpy2.robjects as robjects
import rpy2.robjects.numpy2ri

def run_edger(data, groups, sizes, genes):
    robjects.r('''
        library(edgeR)
    ''')
    params = {'group' : groups, 'lib.size' : sizes}
    dgelist = robjects.r.DGEList(data, **params)
    ms = robjects.r.deDGE(dgelist, doPoisson=True)
    tags = robjects.r.topTags(ms, pair=groups, n=len(genes))
    indexes = [int(t) - 1 for t in tags.rownames()]
    pvals = list(tags.r['adj.P.Val'][0])
    assert len(indexes) == len(pvals)
    pvals_w_index = zip(indexes, pvals)
    pvals_w_index.sort()
    assert len(pvals_w_index) == len(indexes)
    return [p for i,p in pvals_w_index]
```

Write output to a csv file

```
def write_outfile(outfile, genes, conditions, work_counts, probs):
    with open(outfile, "w") as out_handle:
        writer = csv.writer(out_handle)
        writer.writerow(["Region"] +
                ["%s count" % c for c in conditions] + ["edgeR p-value"])
        out_info = []
        for i, gene in enumerate(genes):
            counts = [int(work_counts[c][gene]) for c in conditions]
            out_info.append((probs[i], [gene] + counts))
        out_info.sort()
        [writer.writerow(start + [prob]) for prob, start in out_info]
```

Today's Topics

- Why Python for Life Sciences?
- Important Concepts
  - The Central Dogma of Molecular Biology
  - Types of Biological Data
- Common Tasks in Bioinformatics
  - Python Libraries and Tools
  - Example: Gene Expression Analysis
- Key Issues in doing Biological Data Science
  (a computer scientist perspective)

Key Issues in doing Biological Data Sciences

- **Domain knowledge**
  refer to Getting Started Resources on next slide
- **Interoperability between languages (R⇔Python)**
- **Heterogenous data types and formats**
  data types. DNA/RNA/Protein sequences, gene expression, network, ..
  DB formats. e.g. GenBank, EMBL, SWISSPROT/TrEMBL
  sequence analysis formats. e.g. FASTA, MSA, ClustalW, GFF2
  Ref: http://rice.plantbiology.msu.edu/training/Childs_Data_Formats.pdf
- **Large proportion of time spent in data preprocessing and wrangling**
- **Customized software solutions are needed**

Resources for Getting Started