# COMP-5360 / MATH-4100
# Intro Data Science
# Graphs

Alexander Lex
alex@sci.utah.edu

THE UNIVERSITY OF UTAH

[xkcd]

# Applications of Graphs

Without graphs, there would be none of these:`

facebook
December 2010

# Biological Networks

Interaction between genes, proteins and chemical products

The brain: connections between neurons

Your ancestry: the relations between you and your family

Phylogeny: the evolutionary relationships of life

[Beyer 2014]

CITRATE
ATP
MALONATE, OXALO-ACETATE
SUCCINATE
FUMARATE
SUCCINATE DEHYDROGENASE
FADH₂·E
FAD·E
ACCEPTOR (UBIQUI-NONE)
ACC·H₂ (UBIQUI-NOL)
CoA-SH
SUCCINYL-CoA HYDROLASE
PPi, ATP
H₂O
ADP
GTP(ITP)
GDP(IDP)
SUCCINYL-CoA-LIGASES
SUCCINYL-CoA
PROTEINS
ACETYL-CoA
GLYCYL-tRNA
GLYCINE tRNA LIGASE
GLYCINE
PyrP
5-AMINOLEVULINATE SYNTHASE
increased in acute intermittent porp
5-AMINO-LEVULINATE SYNTHAS
HEME
hν (in plants and photo-bacteria)
NADH + H⁺
FAD·E
DIHYDRO-LIPOAMIDE SUCCINYL TRANS-FERASE
DIHYDRO-LIPO-AMIDE DEHY-DRO-GENASE
Deficient in hyper-oxaluria I
PYRUVATE
ALANINE-GLYOXYLATE TRANS-AMINASE
NADH
FORMATE DEHYDRO-GENASE (CYTOCHROME)
NADH + H⁺
FORMATE DEHYDRO-GENASE
NAD⁺
MOLYBDO-PTERIN
2 Cytb-Fe⁺⁺
FORMYL-CoA
FORMYL-CoA HYDRO-LASE
FORMATE
OXALATE DECARBOXY-LASE
OXALYL-CoA DECARBOXY-LASE
OXALATE-CoA-TRANSFERASE
FERRE-DOXIN (red)
FERRE-DOXIN (ox.)
OXO-GLUTARATE SYNTHASE

Michal 2000

# Graph Analysis Case Study

# Graph Theory Fundamentals

See also “Network Science”, Barabasi
http://barabasi.com/networksciencebook/chapter/2

Tree

Network

Hypergraph

Bipartite Graph

# Königsberg Bridge Problem (1736)

Can you take a walk and visit every land mass without crossing a bridge twice?

Leonhard Euler:
Only possible with a graph with at most two nodes with an odd number of links.
This graph has four nodes with odd number of links.

http://barabasi.com/networksciencebook/chapter/2#bridges

# Graph Terms

A graph **G(V,E)** consists of a set of **vertices V** (also called nodes) and a

set of **edges E** (also called links) connecting these vertices.

**Graph** and **Network** are often used interchangeably

# Graph Term: Simple Graph

A simple graph G(V,E) is a graph which contains **no multi-edges** and **no loops**

Not a simple graph!
→ A ***general graph***

# Graph Term: Directed Graph

A directed graph (digraph) is a graph that discerns between the edges

and

.

# Graph Terms: Hypergraph

A hypergraph is a graph with edges connecting any number of vertices.

Hypergraph Example

# Unconnected Graphs, Articulation Points

***Unconnected graph***

An edge traversal starting from a given vertex cannot reach any other vertex.

Unconnected Graph

***Articulation point***

Vertices, which if deleted from the graph, would break up the graph in multiple sub-graphs.

Articulation Point (red)

# Biconnected, Bipartite Graphs

***Biconnected graph***

A graph without articulation points.

Biconnected Graph

***Bipartite graph***

The vertices can be partitioned in two independent sets.

Bipartite Graph

# Tree

**A graph with no cycles - or:**

**A collection of nodes**

- **contains a root node and 0-n subtrees**
- **subtrees are connected to root by an edge**

# Ordered Tree

≠

# Degree

**Node degree deg(x)**

The number of edges being incident to this node. For directed graphs indeg/outdeg are considered separately.

**Average degree**

$$\langle k \rangle = \frac{1}{N} \sum_{i=1}^{N} k_i = \frac{2L}{N}$$

**Degree distribution**

# Degree Distribution of a real Netw

Protein Interaction Network

# Degrees

Degree is a measure of local importance

# Betweenness Centrality

a measure of how many shortest paths pass through a node

good measure for the overall relevance of a node in a graph

# Degree vs BC

Histogram of Degrees
40
35
30
25
20
15
10
5
0
0
5
10
15
20
25
30
35
40

Histogram of Betweennes Centrality Values
80
70
60
50
40
30
20
10
0
0.0
0.1
0.2
0.3
0.4
0.5
0.6

# Paths & Distances

Path is **route along links**

Path length is the **number of links** contained

Shortest paths connects nodes i and j with the smallest number of links

**Diameter of graph G**
The longest shortest path within G.

A path from 1 to 6

Shortest paths (two) from 1 to 7.

# Graph and Tree Visualization

# Setting the Stage

How to decide which **representation** to use for which **type of graph** in order to achieve which kind of **goal**?

# Different Kinds of Tasks/Goals

Two principal types of tasks: **attribute-based (ABT)** and **topology-based (TBT)**

**Localize** – find a single or multiple nodes/edges that fulfill a given property

- ABT: Find the edge(s) with the maximum edge weight.
- TBT: Find all adjacent nodes of a given node.

**Quantify** – count or estimate a numerical property of the graph

- ABT: Give the number of all nodes.
- TBT: Give the degree of a node.

**Sort/Orde**r – enumerate the nodes/edges according to a given criterion

- ABT: Sort all edges according to their weight.
- TBT: Traverse the graph starting from a given node.

list adapted from Schulz 2010

# Three Types of Graph Representations

Explicit
(Node-Link)

Matrix

Implicit

# Explicit Graph Representations

Node-link diagrams: vertex = point, edge = line/arc

**Free**

**Styled**

**Fixed**

HJ Schulz 2006

# Criteria for Good Node-Link Layout

Minimized **edge crossings**

Minimized **distance** of neighboring nodes

Minimized **drawing area**

Uniform edge **length**

Minimized edge **bends**

Maximized **angular distance** between different edges

Aspect ratio about 1 (not too long and not too wide)

**Symmetry**: similar graph structures should look similar

list adapted from Battista et al. 1999

# Conflicting Criteria

Minimum number of edge crossings

vs.

Uniform edge length

Space utilization

vs.

Symmetry

Schulz 2004

# Force Directed Layouts

Physics model:
edges = springs,
vertices = repulsive magnets

Expander
(pushing nodes apart)

Spring Coil
(pulling nodes together)

# Algorithm

Place Vertices in random locations

While not equilibrium

- calculate force on vertex
  - sum of
    - pairwise repulsion of all nodes
    - attraction between connected nodes
- move vertex by c * force on vertex

# Properties

Generally good layout

Uniform edge length

Clusters commonly visible

Not deterministic

Computationally expensive: $O(n^3)$

$n^2$ in every step, it takes about n cycles to reach equilibrium

Limit (interactive): ~1000 nodes

in practice: damping, center of gravity

http://bl.ocks.org/steveharoz/8c3e2524079a8c440df60c1ab72b5d03

# Giant Hairball

# Graphs in 3D

Why, why not visualize graphs in 3D?

Why, why not use AR/VR?

https://twitter.com/alexsigaras/status/860560655031685121

# Styled / Restricted Layouts

Circular Layout

Node ordering

Edge Clutter

ca. 3% of all possible edges

ca. 6,3% of all possible edges

# Reduce Clutter: Edge Bundling

Holten et al. 2006

# Bundling Strength

mbostock.github.com/d3/talk/20111116/bundle.html

Michael Bostock

# Fixed Layouts

Can’t vary position of nodes

Edge routing important

# Aggregation

https://www.youtube.com/watch?v=E1PVTitj7h0

# Explicit Tree Visualization

Reingold–Tilford layout

[http://billmill.org/pymag-trees/](http://billmill.org/pymag-trees/)

# Explicit Representations

Pros:

- is able to depict all graph classes
- can be customized by weighing the layout constraints
- very well suited for TBTs, if also a suitable layout is chosen

Cons:

- computation of an optimal graph layout is in NP
  (even just achieving minimal edge crossings is already in NP)
- even heuristics are still slow/complex (e.g., naïve spring embedder is in O(n3))
- has a tendency to clutter (edge clutter, “hairball”)

# Matrix Representations

# Matrix Representations

Instead of node link diagram, use adjacency matrix

# Matrix Representations

Examples:

HJ Schulz 2007

# Matrix Representations

Well suited for
neighborhood-related TBTs

Not suited for
path-related TBTs

van Ham et al. 2009
Shen et al. 2007

McGuffin 2012

# Order Critical!

# Implicit Layouts for Trees

# Tree Maps

Johnson and Shneiderman 1991

# Squarified Treemaps

Original Algorithm lead to thin slices

Squarified treemaps [Bruls, Huizing, Van Wijk 2000]

Before

After

# Seeing Tree Structure

Unframed

Framed

# Zoomable Treemap

# Software

Mac: GrandPerspective  Windows: Sequoia View

# Example: Interactive TreeMap of a Million Items

Fekete et al. 2002

# Sunburst: Radial Layout

[Sunburst by John Stasko, Implementation in Caleydo by Christian Partl]

# Differences? Pros, Cons?

# Implicit Representations

Pros:

- space-efficient because of the lack of explicitly drawn edges: scale well up to very large graphs
- in most cases well suited for ABTs on the node set
- depending on the spatial encoding also useful for TBTs

Cons:

- can only represent trees
- since the node positions are used to represent edges, they can no longer be freely arranged (e.g., to reflect geographical positions)
- useless to pursue any task on the edges
- spatial relations such as overlap or inclusion lead to occlusion

# Tree Visualization Reference

# Networks and Attributes

Attributes can influence topology

Path can be slow / blocked

- best route when driving depends on traffic
- biological network depends on many factors

# Challenge: Data Scale & Heterogeneity

Large number of values

- Large datasets have more than 500 experiments

Multiple groups/conditions

Different types of data

# Challenge: Supporting Multiple Tasks

**Two central tasks:**

- **Explore topology of network**
- **Explore the attributes of the nodes (experimental data)**

**Need to support both!**

| | Sample 1 | Sample 2 | Sample 3 |
|---|---|---|---|
| Gene 1 | 1 | 1.1 | 0.4 |
| Gene 2 | 2 | 0.5 | 1.2 |
| Gene 3 | 1.4 | 0.2 | 0.5 |
| Gene 4 | 0.3 | 0.5 | 0.7 |

# Many Node Attributes

| Node | Sample 1 | Sample 2 | Sample 3 | ... |
|---|---|---|---|---|
| A | 0.55 | 0.95 | 0.83 | ... |
| B | 0.12 | 0.42 | 0.16 | ... |
| C | 0.33 | 0.65 | 0.38 | ... |
| ... | ... | ... | ... | |

| Node | Sample 1 | Sample 2 | Sample 3 | ... |
|---|---|---|---|---|
| A | low | low | very high | ... |
| B | normal | low | high | ... |
| C | high | very low | normal | ... |
| ... | ... | ... | ... | |

**How to visualize attribute data on networks?**

# Good Old Color Coding

| | | | | |
|---|---|---|---|---|
| A | -3.4 | 4.2 | 5.1 | 4.2 |
| B | 2.8 | 1.8 | 1.3 | 1.1 |
| C | 3.1 | -2.2 | 2.4 | 2.2 |
| D | -3 | -2.8 | 1.6 | 1.0 |
| E | 0.5 | 0.3 | -1.1 | 1.3 |
| F | 0.3 | 0.3 | 1.8 | -0.3 |

[Lindroos2002]

# Node Attributes

Coloring

Glyphs

-> Limited in scalability

# Small Multiples

Cerebral [Barsky, 2008]

Each dimension in its own window

# Data-driven node positioning

GraphDice

Nodes are laid out according to attribute values

[Bezerianos et al, 2010]

# Path Extraction: enRoute

# enRoute

Melanogenesis
MELANOGENESIS
Melanocyte
αMSH
ASP
MC1R
Gs
AC
cAMP
PKA
CREB
CBP
DNA
MITF
Wnt signaling pathway
Wnt
Frizzled
Go/Gq
Dvl
GSK-3β
β-catenin
TCF/LEF
MAPK signaling pathway
SCF
c-Kit
Ras
Raf
MEK
ERK1/2
Melanoma
TYR
TYRP1
DCT
ER
CAM
CAMK
ET1
ETB-R
Gi/o
PLC
IP3
DG
PKC
Melamin synthesis
PKC-β
Transfer to keratinocytes
Tyrosine
Tyrosine metabolism
Melanin
Melanosome
04916 10/16/12
(c) Kanehisa Laboratories
Pancreatic cancer
KRAS
ARAF
MAP2K1
MAPK8
ErbB signaling pathway
GRB2
SOS1
HRAS
ARAF
MAP2K1
MAPK1
MYC
Selected Path
pancreas
pancreas
Melanogenesis
ErbB signaling pathway
KITLG
KIT
HRAS
RAF1
MAP2K1
MAPK1
MYC
NRAS
RAF1
BRAF
ARAF
MAP2K2
MAP2K1
MAPK1
MAPK3
MYC

File Data Window View Help

Entourage

Pathways

Pathway

Filter:
<None>

1 C donor
2-Oxocarboxylic acid
ABC transporters
ABC-family proteins
ACE Inhibitor Pathwa
Acetylcholine Synthes
Acute myeloid leukem
Adherens junction
Adipocyte TarBase
Adipocytokine signali
Adipogenesis
Advanced glycosylatio
Aflatoxin B1 metaboli
African trypanosomias
AGE/RAGE pathway
AhR pathway
Alanine and aspartate
Alanine, aspartate an
Alcoholism
Aldosterone-regulated
Allograft rejection
Allograft rejection
Alpha 6 Beta 4 signal
alpha-Linolenic acid
Alzheimer's disease
Alzheimers Disease
amino acid conjugatio
amino acid conjugatio
Amino sugar and nucl
Aminoacyl-tRNA bios
Amoebiasis
Amphetamine addicti
AMPK signaling
Amyotrophic lateral sc
Androgen receptor si
Angiogenesis
Angiogenesis
angiogenesis overvie
Antigen processing an
APC/C-mediated degra
Apoptosis
Apoptosis
Apoptosis Meta Path
Apoptosis Modulation
Apoptosis Modulation
Apoptosis, anoikis an

Selected Path

Toggle summary button
Summary of all dimensions
All dimensions shown separately
Not Mutated
Mutated
Not Mutated
Mutated
Race
Days to Deat
Gender
Context (non-genetic) datasets
EGF
EGFR
PLCG1
PLCG2
PRKCB1
PRKCA
PRKCG
NRAS
Dimensions sorted by this row
Multi-mapping of gene family
Collapsed incoming node
Expanded outgoing nodes
Glioma
HRAS
ARAF
PIK3R5
Not Mutated
Mutated
Not Mutated
Mutated
Groups of same dataset
Different Datsets

# Case Study: CCLE Data

Pancreatic cancer
Non-small cell lung cancer
Proliferation
Survival
Angiogenesis
Adhesion
Migration / Invasion
Differentiation
Glioma
DNA
Jun
Elk
Myc
+p
ERBB2
GRB2
SOS1
HRAS
ARAF
MAP2K1
MAPK1
CCND1
PIK3R5
AKT3
FOXO3
CALML6
CAMK2A
ErbB signaling pathway
SHC2
Erlotinib
Lapatinib
SHC4
SHC3
SHC1
SOS2
NRAS
breast

[EuroVis '16]

Honorable Mention Award

# Pathfinder: Visual Analysis of Paths in Graphs

**Intelligence Data:** How are two suspects connected?

**Intelligence Data:** How are two suspects connected?

**Biological Network:** How do two genes interact?

**Coauthor Network:** How is HP Pfister connected to Ben Shneiderman?

# Pathfinder

## Visual Analysis of Paths
## n Large Multivariate Graphs

# Pathfinder Approach

Query for paths

# Pathfinder Approach

Shows query-result diagram.

# Pathfinder Approach

Update ranking to identify important paths

Path Score

1.

2.

# Pathfinder Approach

Update ranking to identify important paths

Pathfinder

Start End Advanced Query Length Paths

Path Statistics

Path List

Path Topology

Active Page All

# Query Interface

# Path Representation

# Pathways

# Grouped Copy Number and Gene Expression Da

# Visualizing Edge Attributes

Most common ways to encode edge attributes

**Quantitative:** Width

**Ordinal:** Saturation

**Nominal:** Style

# Visualizing Edge Attributes

In practice very limited

Example: Sashimi Plots

# What's the Problem?

# Junction View

# Junction View - Group Comparison

# Junction View - Group Comparison

# Junction View - Group Comparison

# Case Study: Leukemia vs Glioblastoma

# Graph Tools & Applications

# Gephi

http://gephi.org

Gephi
makes graphs handy

Download | Blog | Wiki | Forum | Support | Bug tracker

Home | Features | Plugins | Users | Developers | Partners

## The Open Graph Viz Platform

**Gephi is a visualization and exploration platform for all kinds of networks and complex systems, dynamic and hierarchical graphs.**

**Runs on Windows, Linux and Mac OS X. Gephi is open-source and free.**

Learn More on Gephi Platform »

Download FREE
Gephi 0.7 alpha

Release Notes | System Requirements

- Features
- Quick start
- Screenshots
- Videos

**Gephi has been accepted again for Google Summer of Code! The program is the best way for students around the world to start contributing to an open-source project. Students, apply now for Gephi proposals. Come to the GSOC forum section and say Hi! to this topic.**

**Learn More »**

# Cytoscape

http://www.cytoscape.org/

Open source pla

# Cytoscape Web

http://cytoscapeweb.cytoscape.org/

# NetworkX

https://networkx.github.io/

## NetworkX

NetworkX Home | Documentation | Download | Developer (Github)

### High-productivity software for complex networks

NetworkX is a Python language software package for the creation, manipulation, and study of the structure, dynamics, and functions of complex networks.

**Documentation**
*all documentation*

**Examples**
*using the library*

**Reference**
*all functions and methods*

### Features

- Python language data structures for graphs, digraphs, and multigraphs.
- Nodes can be "anything" (e.g. text, images, XML records)
- Edges can hold arbitrary data (e.g. weights, time-series)
- Generators for classic graphs, random graphs, and synthetic networks
- Standard graph algorithms
- Network structure and analysis measures
- Open source BSD license
- Well tested: more than 1800 unit tests, >90% code coverage
- Additional benefits from Python: fast prototyping, easy to teach, multi-platform

**Versions**

**Latest Release**

1.8.1 - 4 August 2013
downloads | docs | pdf

**Development**

1.9dev
github | docs | pdf
build passing
coverage 83%

**Contact**

Mailing list
Issue tracker
Developer guide