PRINCIPLES OF DATA SCIENCE

JOHN P DICKERSON

Lecture #7 – 10/10/2018

CMSC641
Wednesdays
7:00pm – 9:30pm

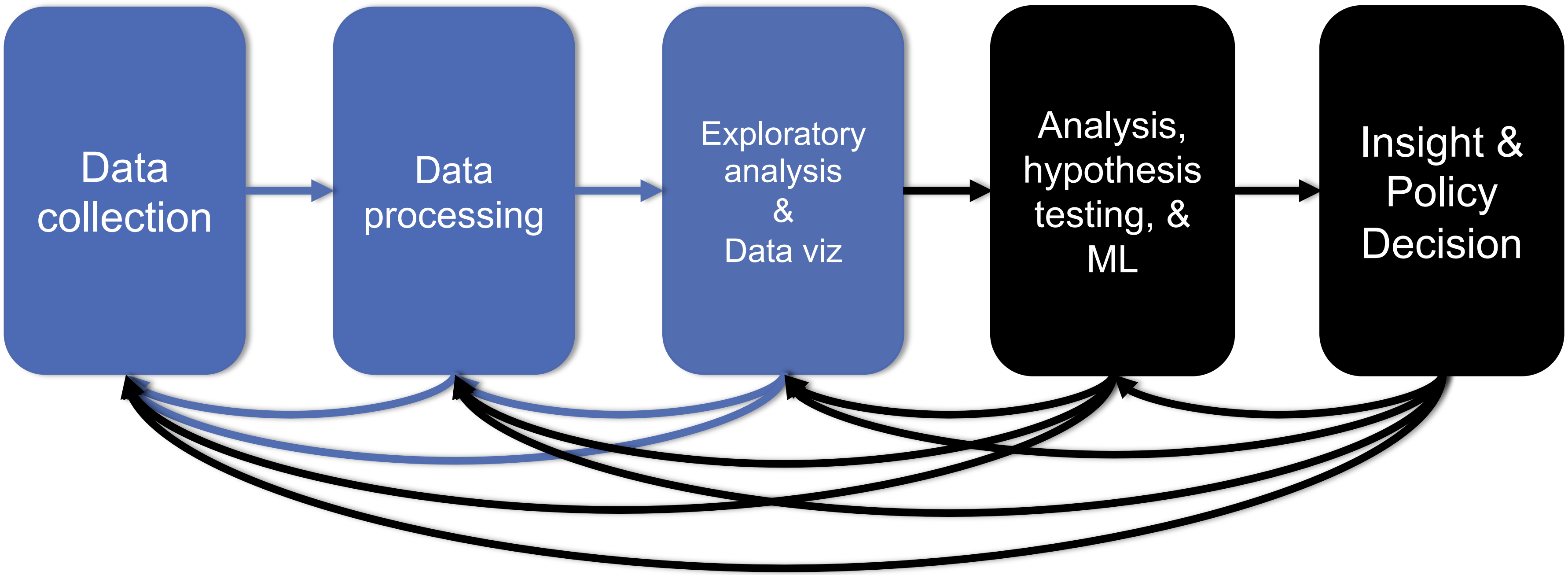
COMPUTER SCIENCE
UNIVERSITY OF MARYLAND

ANNOUNCEMENTS

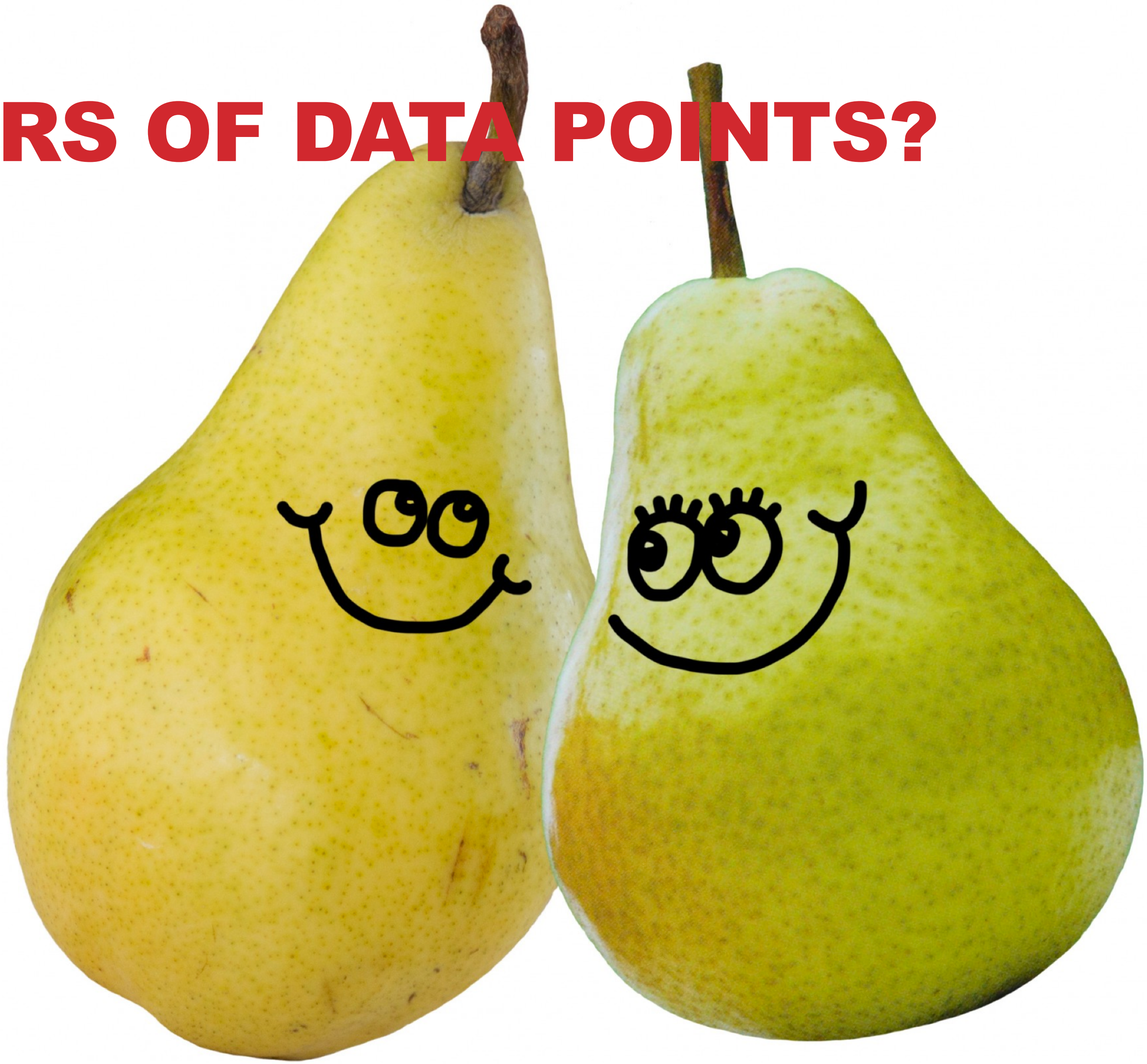
Mini-Project #2 is out!

- It is linked to from ELMS; also available at: https://github.com/umddb/cmsc641-fall2018/tree/master/project2
- Deliverable is a .ipynb file submitted to ELMS
- Due Wednesday, October 24th

WRAP-UP FROM LAST LECTURE …



PAIRS OF DATA POINTS?



VARIANCE & STDEV: UNIVARIATE MEASURES OF DISPERSION

Variance = s_X^2 = $\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$ or $\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$

Standard deviation = s_X = $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2}$

The variance is commonly used statistic for spread

- What are the units of the variance ??????????

Standard deviation "fixes this," can be used as an interpretable unit of measurement

Why n-1? http://nebula.deanza.edu/~bloom/math10/m10divideby_nminus1.pdf

VARIANCE, ASIDE: WHY DIVIDE BY N-1?

Remember: we are typically calculating the mean / median / variance / etc of a sample of a population

- Want that {mean, median, variance, …} to be an "unbiased" estimate of the true population's {mean, median, variance, …}

Unbiased? Consider variance …

1. Look at every possible sample of the population
2. Compute sample variance of each population
3. Is the average of those variances equal to the population variance? If so, then this is an "unbiased" estimator.

VARIANCE, ASIDE: WHY DIVIDE BY N-1?

Dividing by n-1 in the sample variance computation leads to an unbiased estimate of the population variance

$$\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2$$

Intuition. Fix a sample …

- Variance measures distribution around a mean
- Sampled values are, on average, closer to sample mean than to true population mean
- So, we will underestimate the true variance slightly
- Using n-1 instead of n makes our variance calculation bigger

This "embiggening" impacts smaller *n* more than larger *n*

- Larger samples are better estimates of population
- If sample is the population, just divide by *n* …

MULTIVARIATE: CORRELATION

Variables Y and X vary together

Causality vs. correlation: Does movement in X “cause” movement in Y in some metaphysical sense?

Correlation

- Simultaneous movement through a statistical relationship
- Simultaneous variation “induced” by the variation of a common third effect

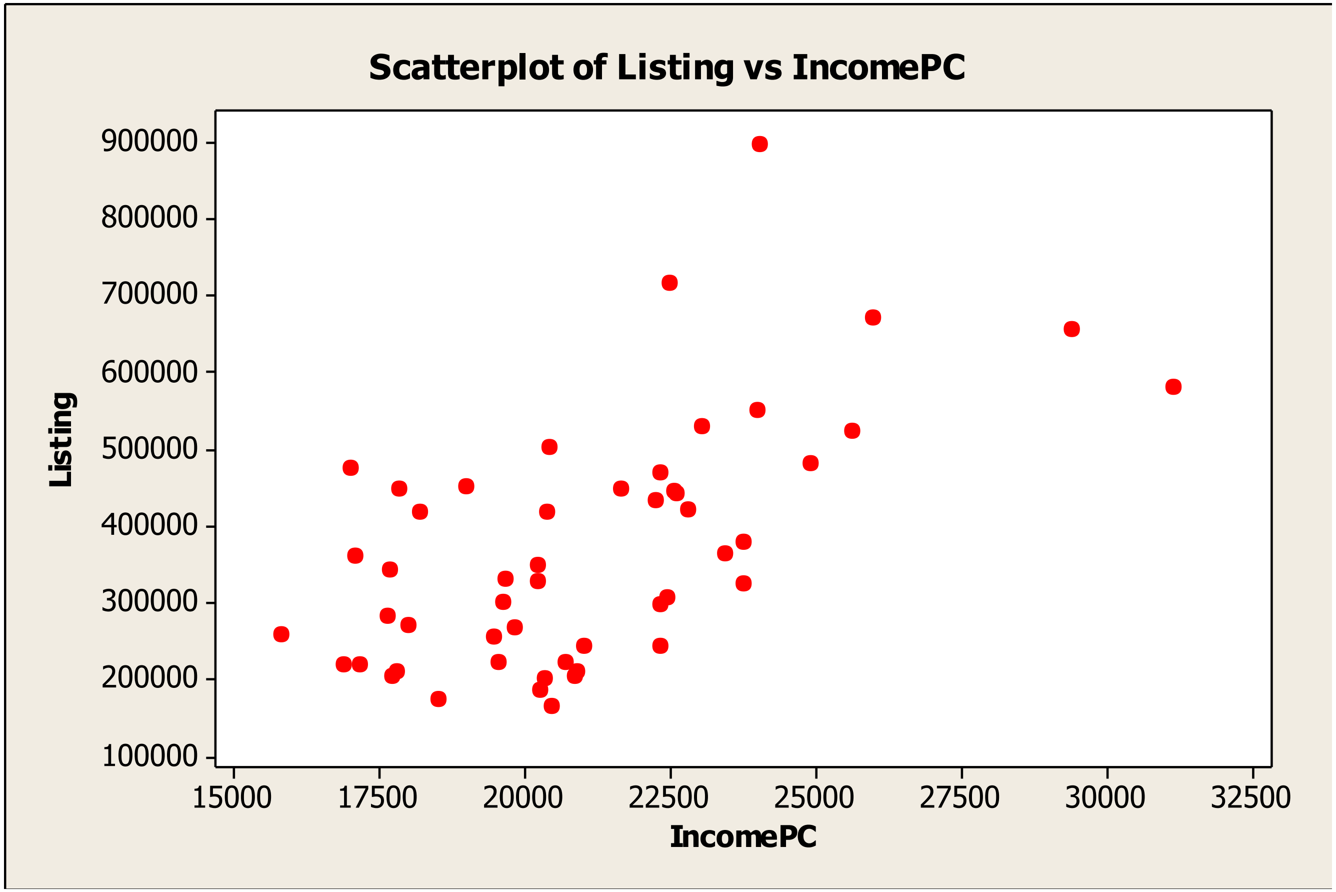
HOUSE PRICES & PER CAPITA INCOME

State	Listing	IncomePC
Hawaii	896800	24057
California	713864	22493
New York	668578	25999
Connecticut	654859	29402
Dist. Columbia	577921	31136
Nevada	549187	24023
New Jersey	529201	23038
Massachusetts	521769	25616
Wyoming	499674	20436
Maryland	480578	24933
Utah	475060	17043
Colorado	467979	22333
Arizona	448791	19001
Florida	447698	21677
Montana	446584	17865
Virginia	443618	22594
Washington	440542	22610

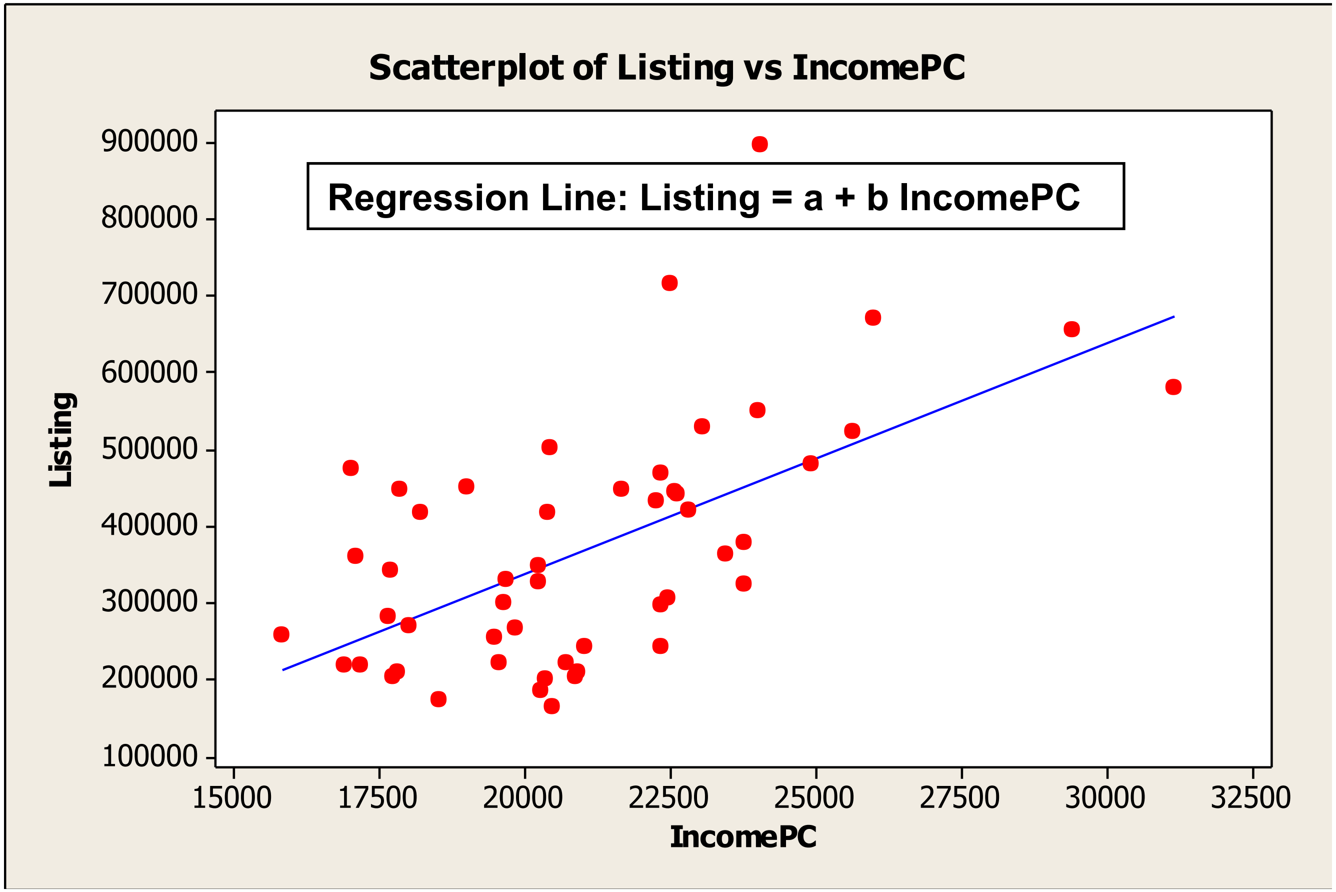
State	Listing	IncomePC
Rhode Island	432534	22251
Delaware	420845	22828
Oregon	417551	20419
Idaho	415885	18231
Illinois	377683	23784
New Hampshire	361691	23434
New Mexico	358369	17106
Vermont	346469	20224
South Carolina	340066	17695
North Carolina	330432	19669
Georgia	326699	20251
Alaska	324774	23788
Minnesota	306009	22453
Maine	299796	19663
Pennsylvania	295133	22324
Louisiana	280631	17651
Alabama	269135	18010

State	Listing	IncomePC
Texas	266388	19857
Mississippi	255774	15838
Tennessee	255064	19482
Wisconsin	243006	21019
Michigan	241107	22333
Missouri	221773	20717
South Dakota	220708	19577
West Virginia	219275	17208
Arkansas	217659	16898
Ohio	209189	20928
Kentucky	208391	17807
Oklahoma	203926	17744
Kansas	201389	20896
Indiana	200683	20378
Iowa	184999	20265
North Dakota	173977	18546
Nebraska	164326	20488

SCATTER PLOT SUGGESTS POSITIVE CORRELATION

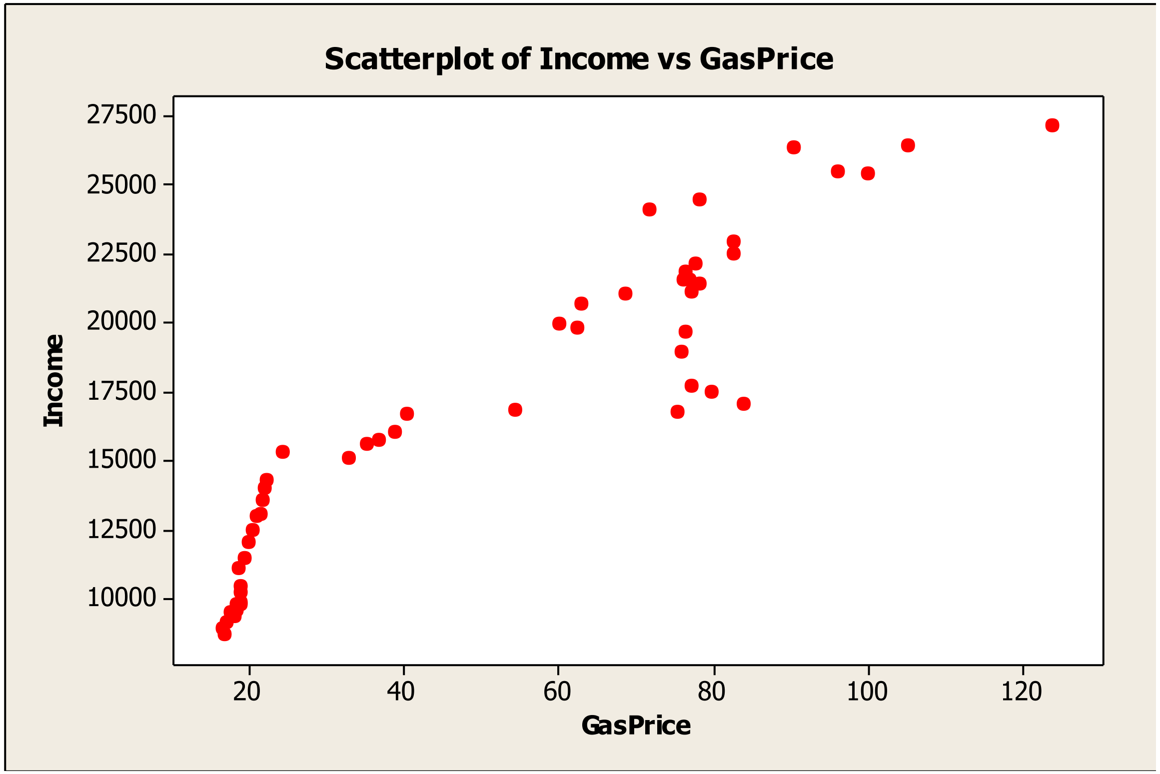


LINEAR REGRESSION MEASURES CORRELATION



CORRELATION IS NOT CAUSATION

Price and income seem to be positively correlated.

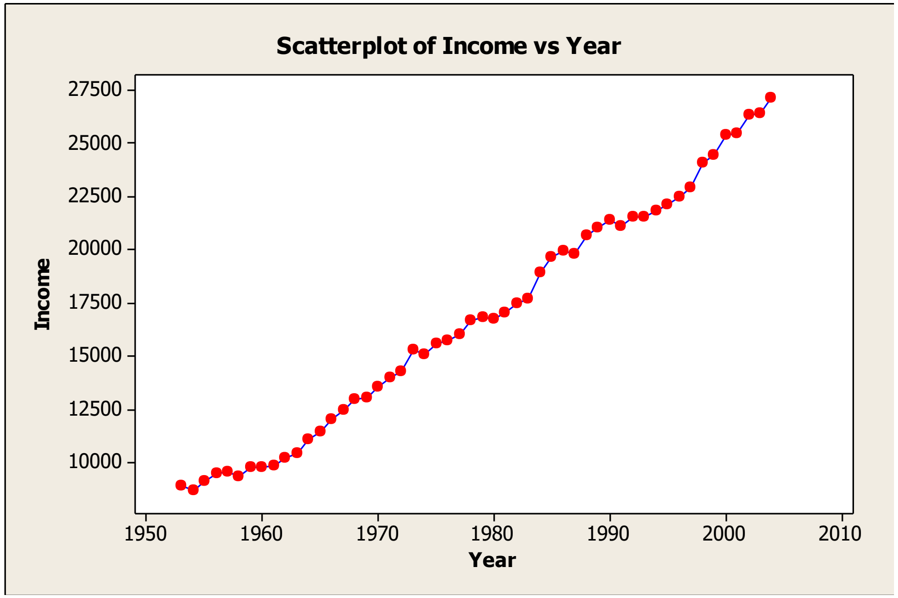


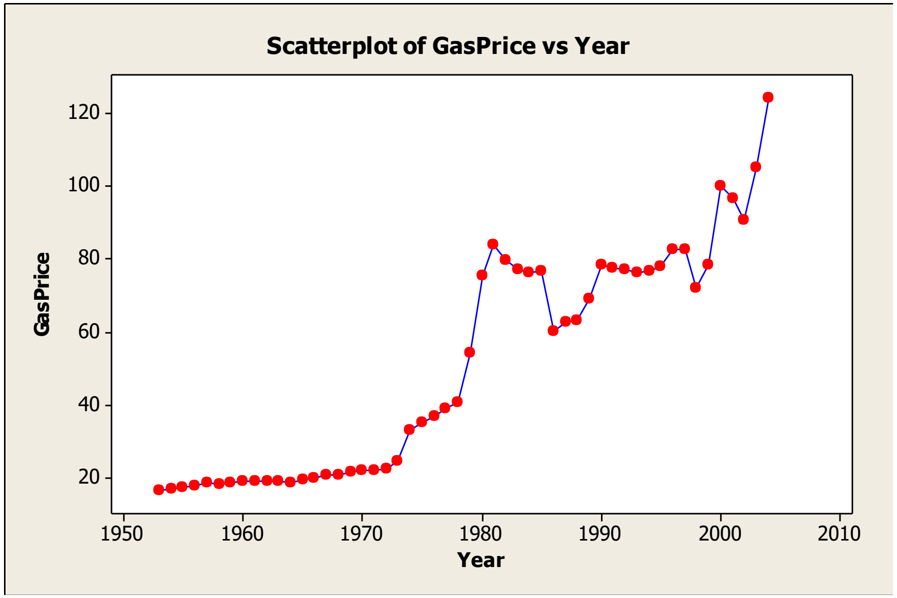
Does a rise in income cause a rise in gas prices ??????????????

US gasoline prices, 1953-2004, plotted against per-capita US income

A HIDDEN RELATIONSHIP

**Not positively "related" to each other;
both positively related to "time."**





"RELATED" ...?

Want to capture: some variable X varies in the same direction and at the same scale as some other variable Y

$$cov(x,y) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$

What happens if:

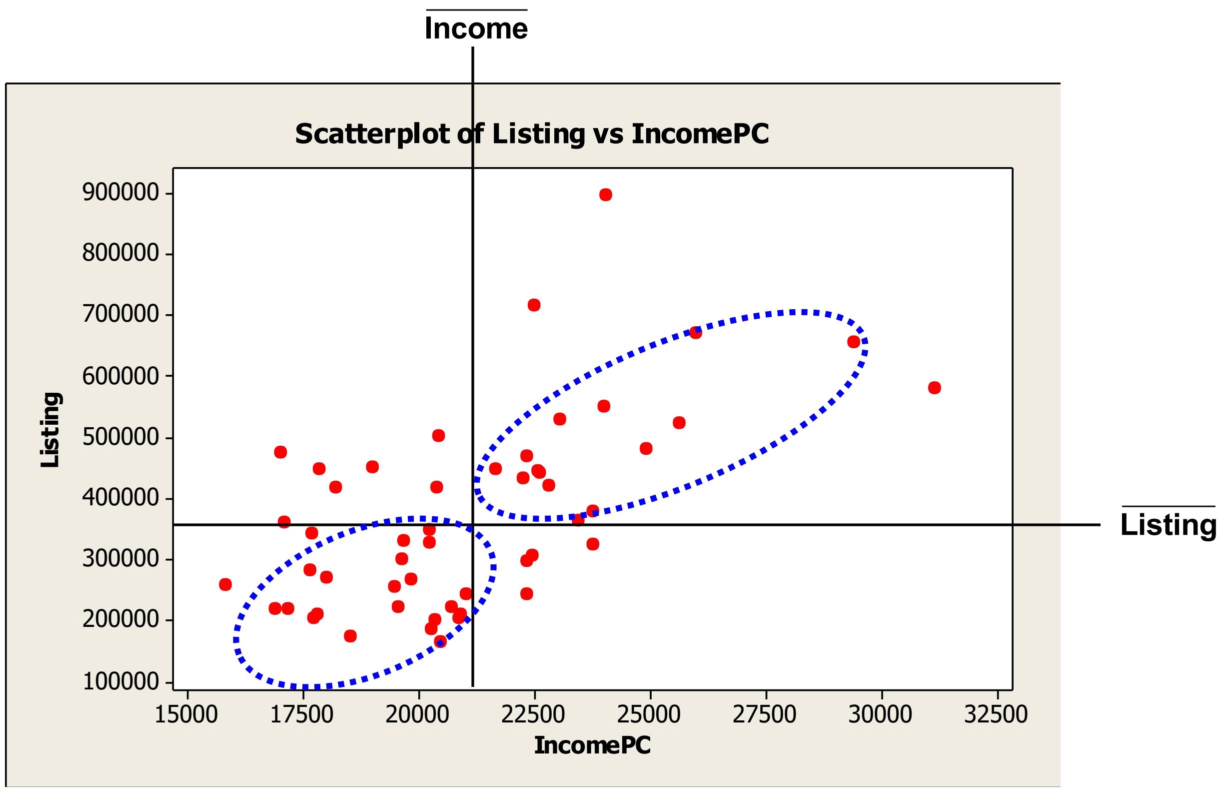
- X varies in the opposite direction as Y ????????
- X varies in the same direction as Y ????????

What are the units of the covariance ????????

Pearson's correlation coefficient is unitless in [-1,+1]:

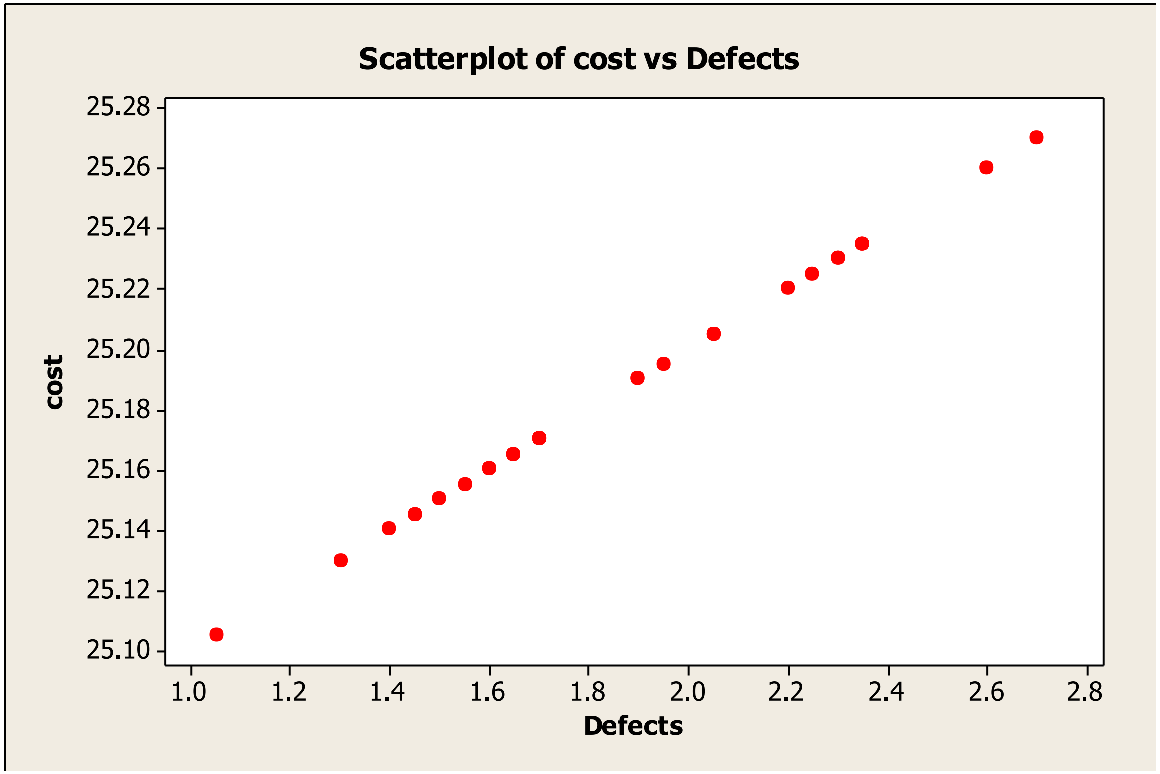
$$cor(x,y) = \frac{cov(x,y)}{sd(x)sd(y)}$$

CORRELATION

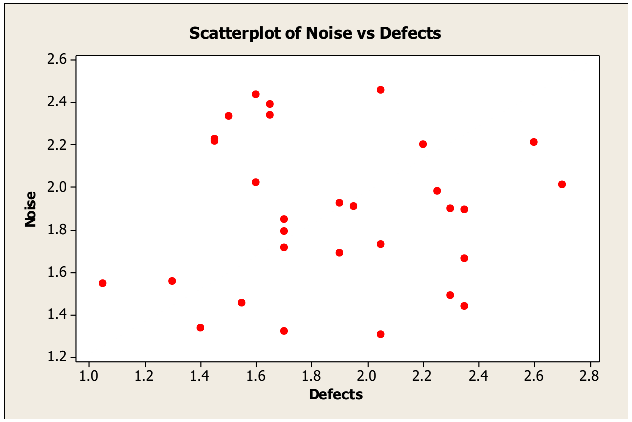


$r_{\text{Income,Listing}} = +0.591$

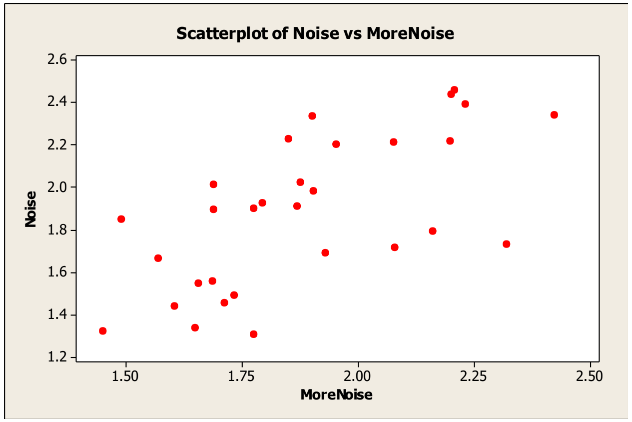
CORRELATIONS



r = +1.0

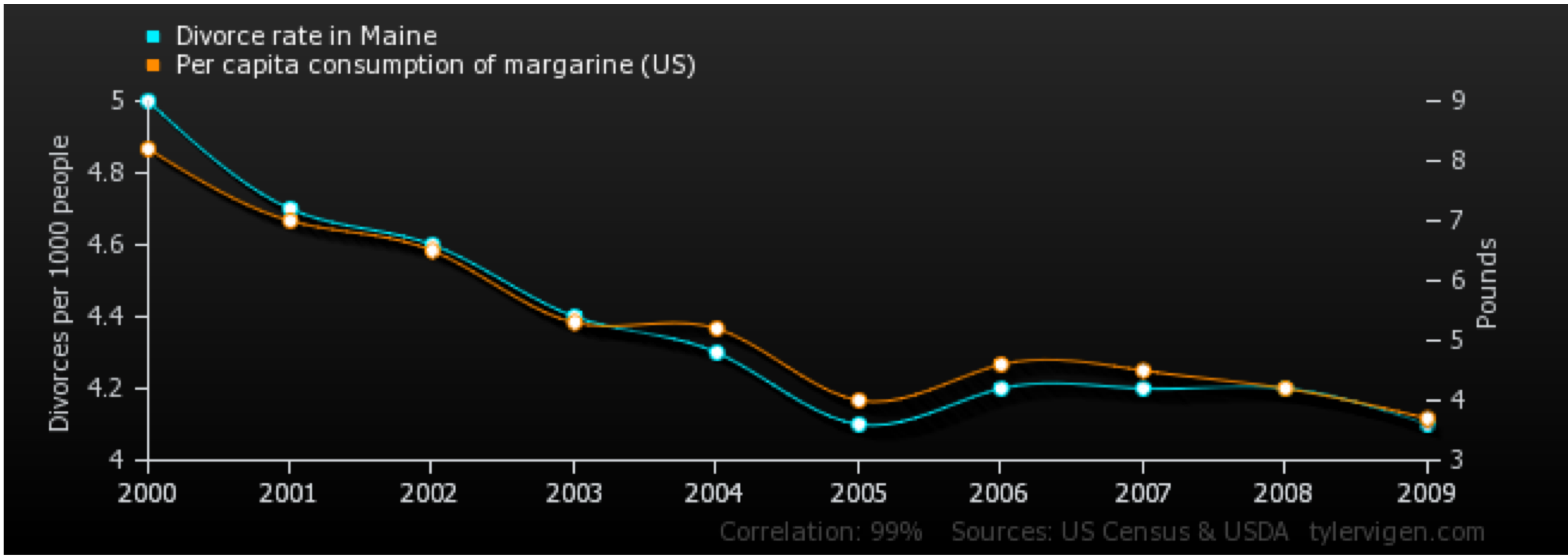


r = 0.0



r = +0.5

CORRELATION IS NOT CAUSATION!!!



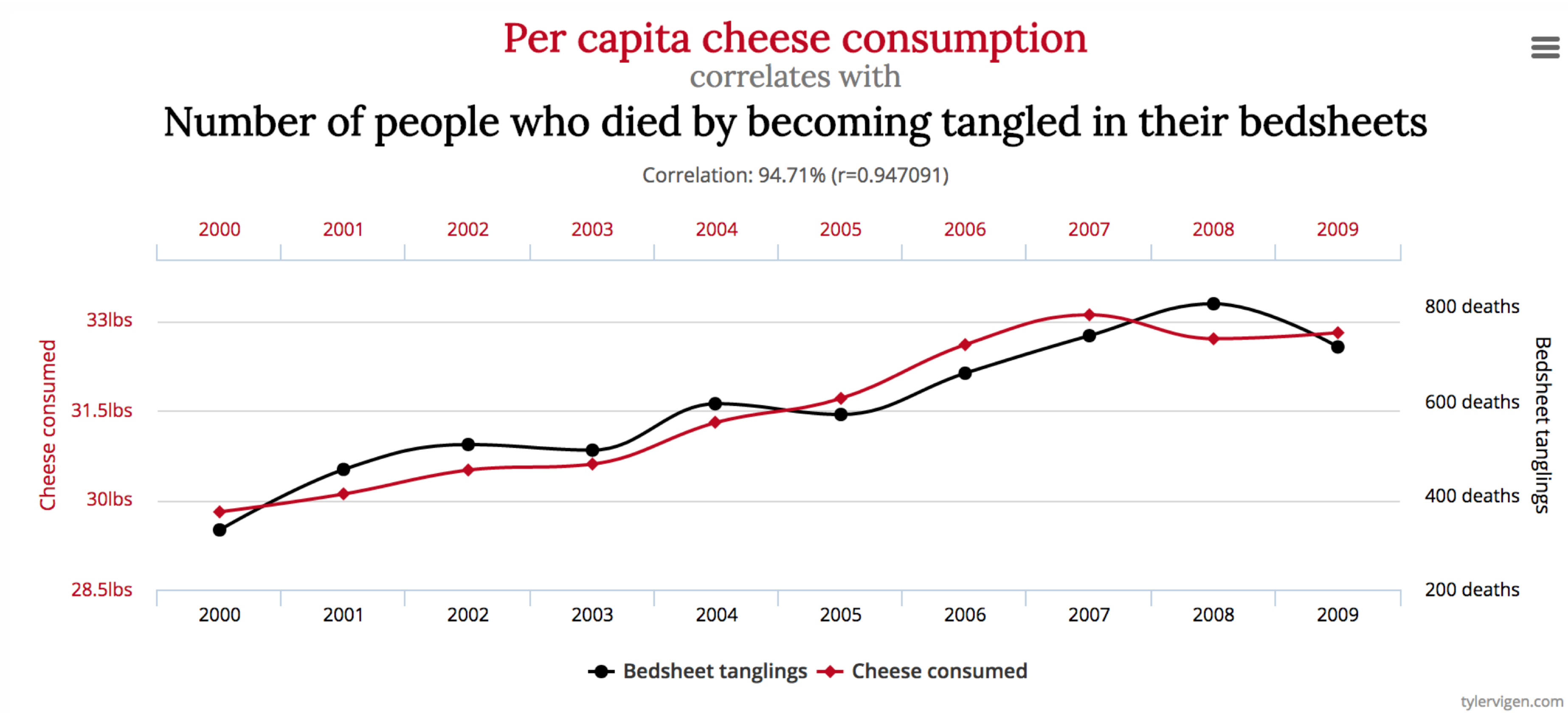
	2000	**2001**	**2002**	**2003**	**2004**	**2005**	**2006**	**2007**	**2008**	**2009**
Divorce rate in Maine Divorces per 1000 people (US Census)	5	4.7	4.6	4.4	4.3	4.1	4.2	4.2	4.2	4.1
Per capita consumption of margarine (US) Pounds (USDA)	8.2	7	6.5	5.3	5.2	4	4.6	4.5	4.2	3.7

r=0.993

?????????

http://tylervigen.com/spurious-correlations

JUST TO DRIVE THE POINT HOME …



Data sources: U.S. Department of Agriculture and Centers for Disease Control & Prevention

TRANSFORMATIONS

TRANSFORMATIONS

So, you've figured out that your data are:

- Skewed
- Have vastly different ranges across datasets and/or different units

What do you do?

Transform the variables to:
- ease the validity and interpretation of data analyses
- change or ease the type of Stat/ML models you can use

STANDARDIZATION

Transforming the variable to a comparable metric

- known unit
- known mean
- known standard deviation
- known range

Three ways of standardizing:

- P-standardization (percentile scores)
- Z-standardization (z-scores)
- D-standardization (dichotomize a variable)

WHEN YOU SHOULD ALWAYS STANDARDIZE

When averaging multiple variables, e.g. when creating a socioeconomic status variable out of income and education.

When comparing the effects of variables with unequal units, e.g. does age or education have a larger effect on income?

Slides adapted from Maarten Buis

P-STANDARDIZATION

Every observation is assigned a number between 0 and 100, indicating the percentage of observation beneath it.

Can be read from the cumulative distribution

In case of knots: assign midpoints

The median, quartiles, quintiles, and deciles are special cases of P-scores.

	rent	cum %	percentile
room 1	175	5,3%	5,3%
room 2	180	10,5%	10,5%
room 3	185	15,8%	15,8%
room 4	190	21,1%	21,1%
room 5	200	26,3%	26,3%
room 6	**210**	31,6%	**36,8%**
room 7	**210**	36,8%	**36,8%**
room 8	**210**	42,1%	**36,8%**
room 9	230	47,4%	47,4%
room 10	**240**	52,6%	**55,3%**
room 11	**240**	57,9%	**55,3%**
room 12	**250**	63,2%	**65,8%**
room 13	**250**	68,4%	**65,8%**
room 14	280	73,7%	73,7%
room 15	**300**	78,9%	**81,6%**
room 16	**300**	84,2%	**81,6%**
room 17	310	89,5%	89,5%
room 18	325	94,7%	94,7%
room 19	620	100,0%	100,0%

P-STANDARDIZATION

Turns the variable into a ranking, i.e. it turns the variable into a ordinal variable.

It is a non-linear transformation: relative distances change

Results in a fixed mean, range, and standard deviation; M=50, SD=28.6, This can change slightly due to knots

A histogram of a P-standardized variable approximates a uniform distribution

CENTERING AND SCALING

Transform your data into a unitless scale

- Put data into "standard deviations from the mean" units
- This is called standardizing a variable, into standard units

Given data points $x = x_1, x_2, \ldots, x_n$:

$$z_i = \frac{(x_i - \bar{x})}{\mathrm{sd}(x)}$$

Translates *x* into a scaled and centered variable *z*

What is the mean of *z* ??????????

What is the standard deviation of z ??????????

CENTERING OR SCALING

Maybe you just want to center the data:

$$z_i = (x_i - \bar{x})$$

What is the mean of *z* ??????????

What is the standard deviation of z ??????????

Maybe you just want to scale the data:

$$z_i = \frac{x_i}{\mathrm{sd}(x_i)}$$

What is the mean of *z* ??????????

What is the standard deviation of z ??????????

DISCRETE TO CONTINUOUS VARIABLES

Some models only work on continuous numeric data

Convert a binary variable to a number ???????????

- health_insurance = {"yes", "no"} → {1, 0}

Why not {-1, +1} or {-10, +14}?

- 0/1 encoding lets us say things like "if a person has healthcare then their income increases by $X."
- Might need {-1,+1} for certain ML algorithms (e.g., SVM)

DISCRETE TO CONTINUOUS VARIABLES

What about non-binary variables?

My main transportation is a {BMW, Bicycle, Hovercraft}

One option: { BMW → 1, Bicycle → 2, Hovercraft → 3 }

- Problems ??????????

One-hot encoding**: convert a categorical variable with N values into a N-bit vector:**

- BMW → [1, 0, 0]; Bicycle → [0, 1, 0]; Hovercraft → [0, 0, 1]

```
# Converts dtype=category to one-hot-encoded cols
cols = ['my_transportation']
df = df.get_dummies( columns = cols )
```

CONTINUOUS TO DISCRETE VARIABLES

Do doctors prescribe a certain medication to older kids more often? Is there a difference in wage based on age?

Pick a discrete set of bins, then put values into the bins

Equal-length bins:

- Bins have an equal-length range and skewed membership
- Good/Bad ????????

Equal-sized bins:

- Bins have variable-length ranges but equal membership
- Good/Bad ????????

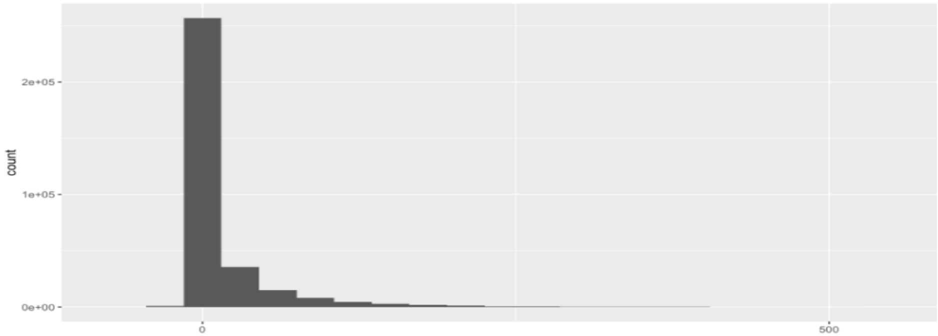
SKEWED DATA

Skewed data often arises in multiplicative processes:

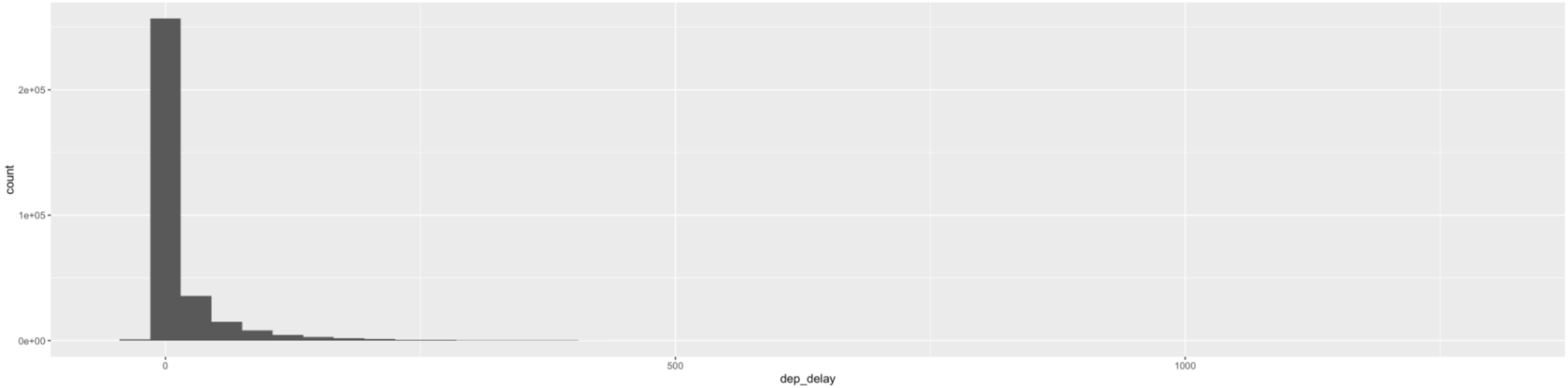
- Some points float around 1, but one unlucky draw → 0

Logarithmic transforms reduce skew:

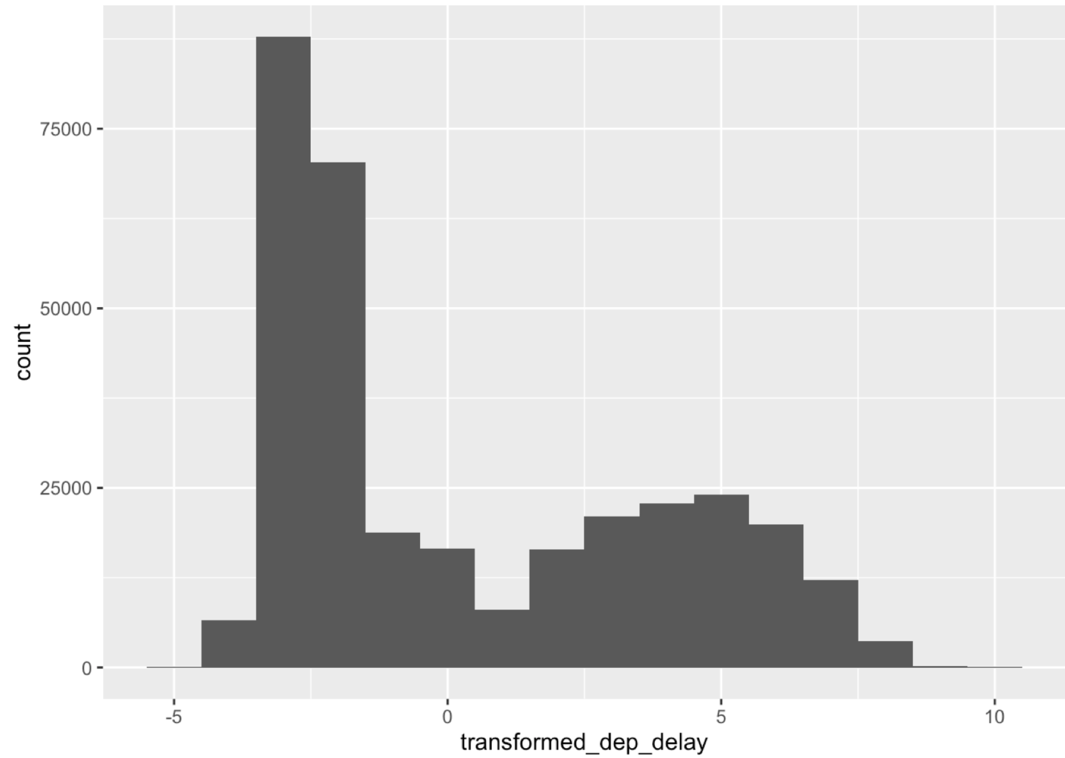
- If values are all positive, apply $\log_2$ transform
- If some values are negative:
  - Shift all values so they are positive, apply $\log_2$
  - Signed log: $\text{sign}(x) * \log_2(|x| + 1)$



SKEWED DATA



log$_2$ transform on airline takeoff delays



NEXT UP:

VISUALIZATION, GRAPHS, & NETWORKS

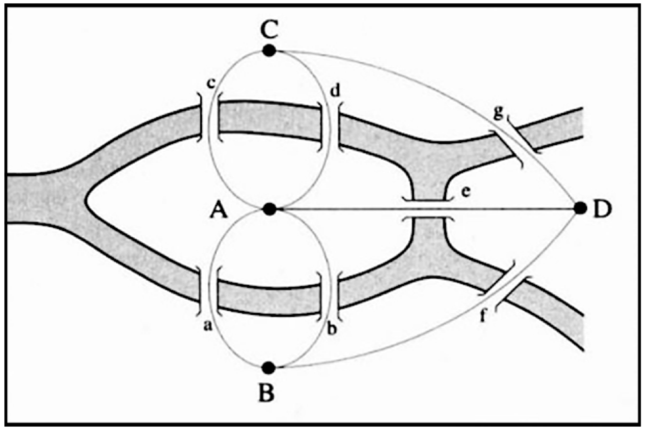
AND NOW!

Graph Processing

- Representing graphs
- Centrality measures
- Community detection

Natural Language Processing

- Bag of Words, TF-IDF, N-grams
- (If we get to this today …)



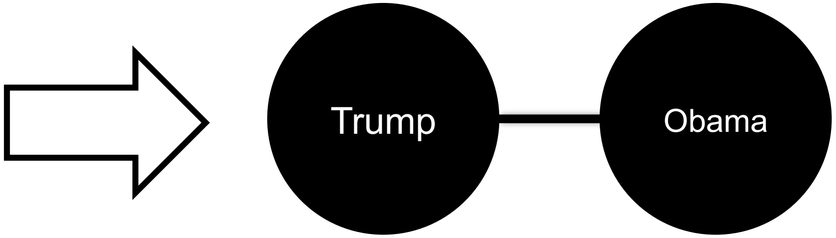
Thank you to: Sukumar Ghosh (Iowa), Lei Tang (Yahoo!), Huan Liu (ASU), Zico Kolter (CMU)

NETWORKS? GRAPHS?

Networks are systems of interrelated objects

Graphs are the mathematical models used to represent networks

In data science, we will use algorithms on graphs to answer questions about real-world networks.

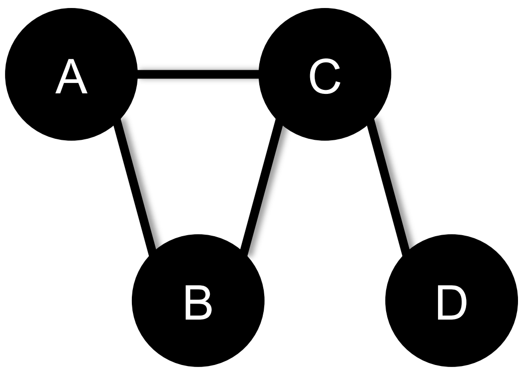


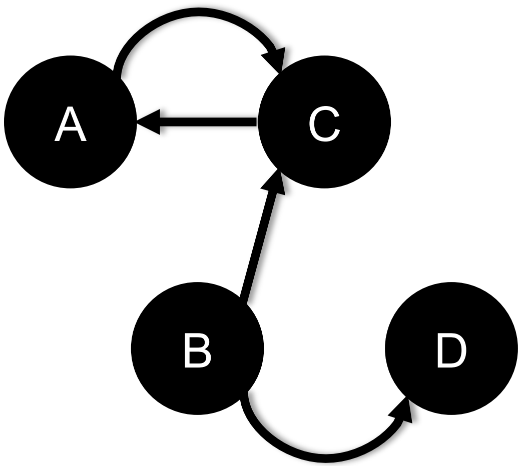
GRAPHS

Nodes = Vertices
Edges = Arcs

A graph G = (V,E) is a set of vertices V and edges E

Edges can be undirected or directed





V = {A, B, C, D}
E = {(A,B), (B,C), (C,D), (A,C)}

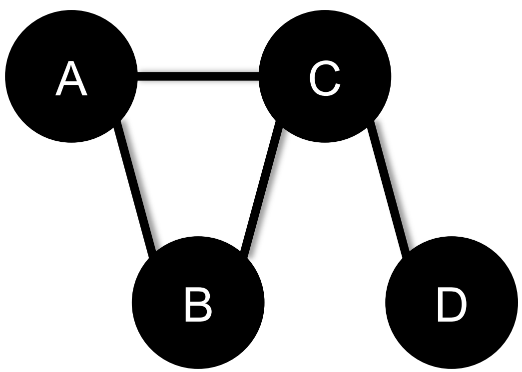
V = {A, B, C, D}
E = {(A,C), (C,A), (B,C), (B,D)}

Examples of directed vs undirected graphs ????????????

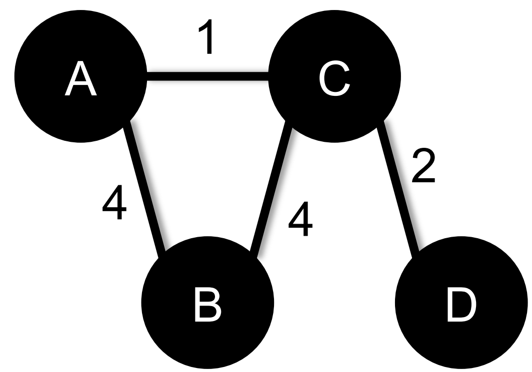
GRAPHS

Edges can be unweighted or weighted

- Unweighted → all edges have unit weight



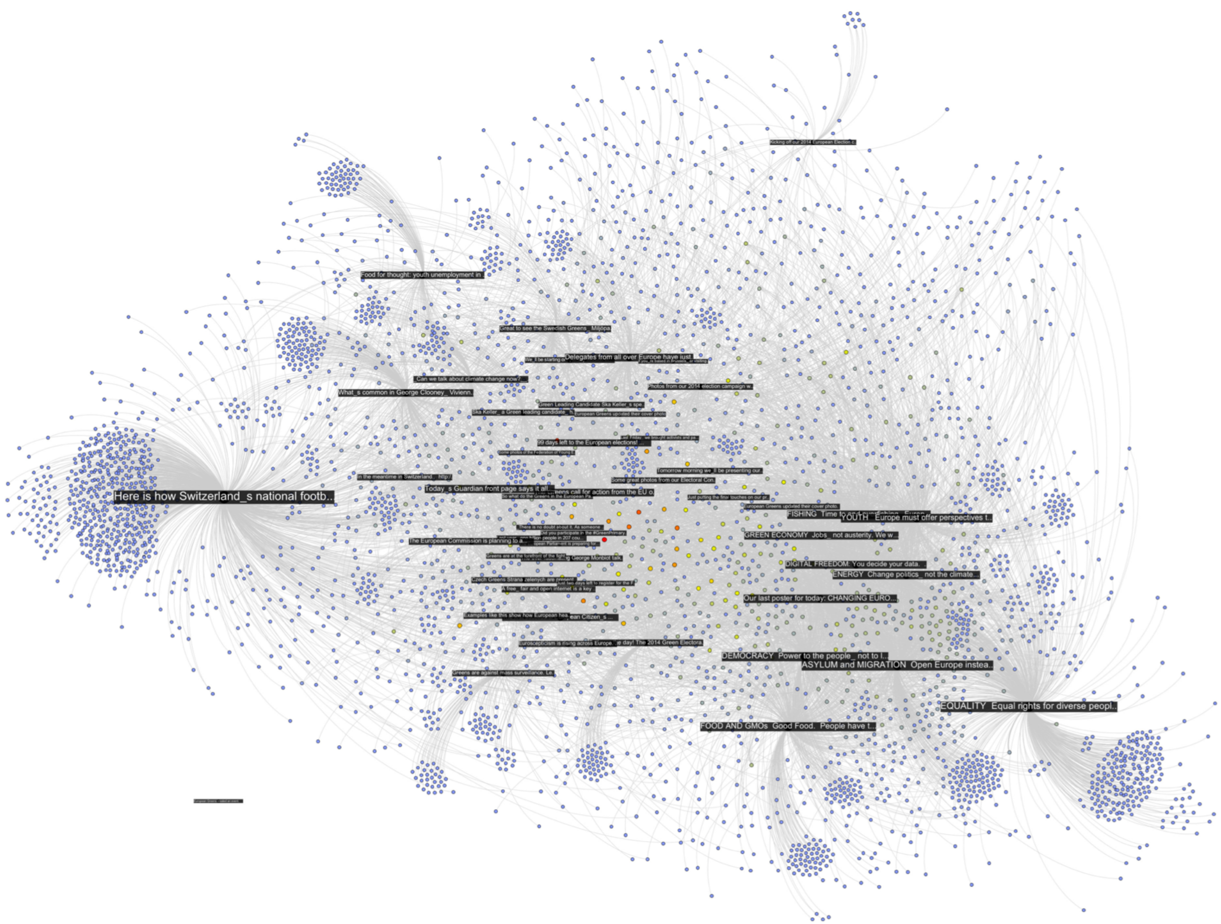
Unweighted



Weighted

Examples of unweighted and weighted graphs ????????????

GRAPHS AND THE NETWORKS THEY REPRESENT

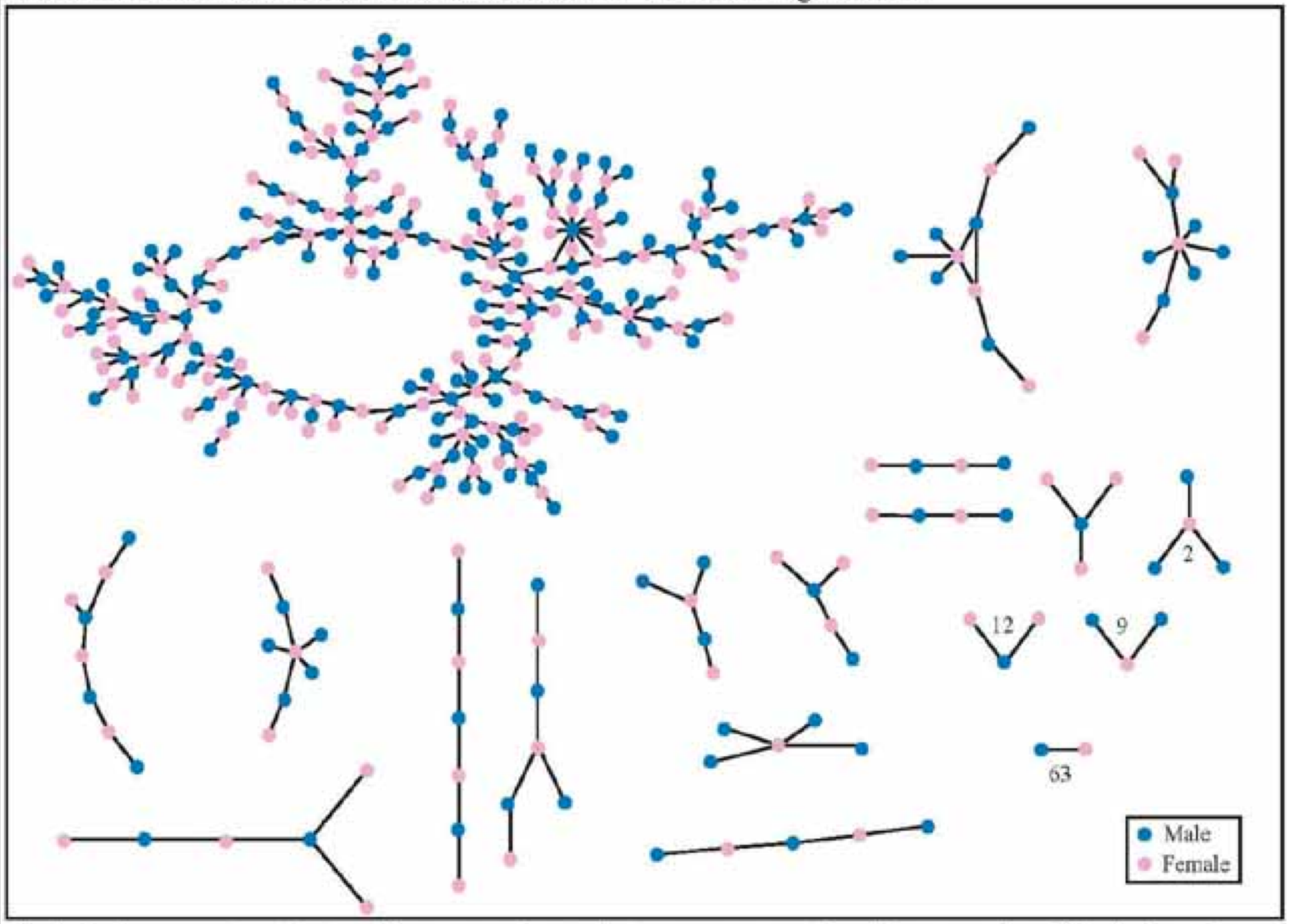


Facebook posts (in black), and users liking or commenting on those posts

http://thepoliticsofsystems.net/category/network-theory/

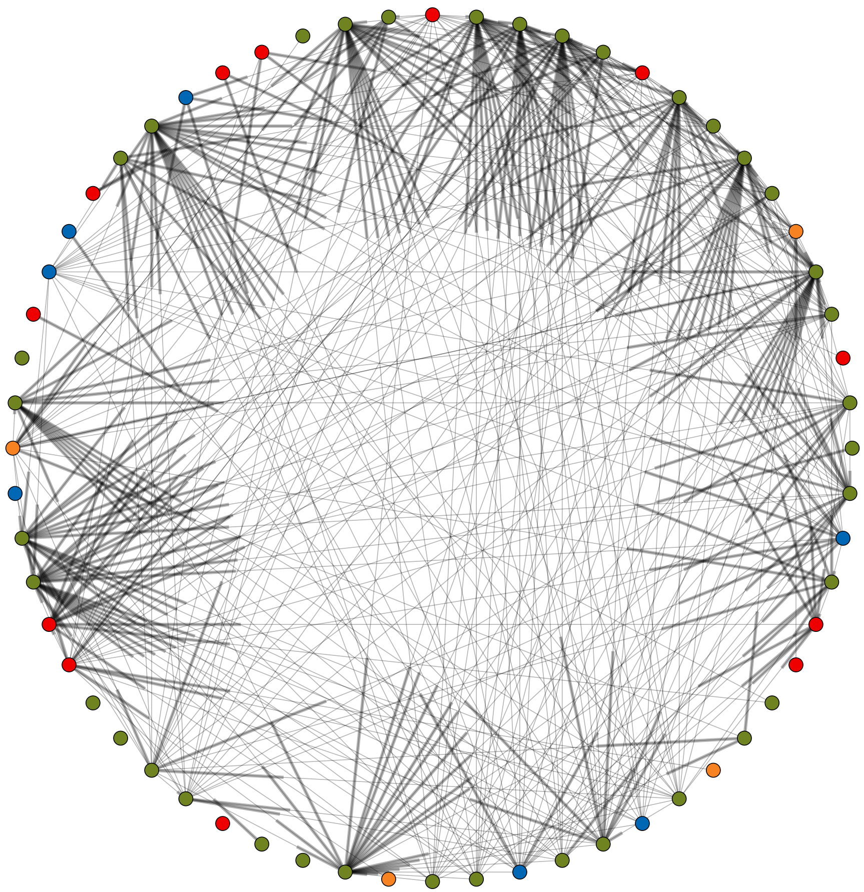
GRAPHS AND THE NETWORKS THEY REPRESENT

The Structure of Romantic and Sexual Relations at "Jefferson High School"

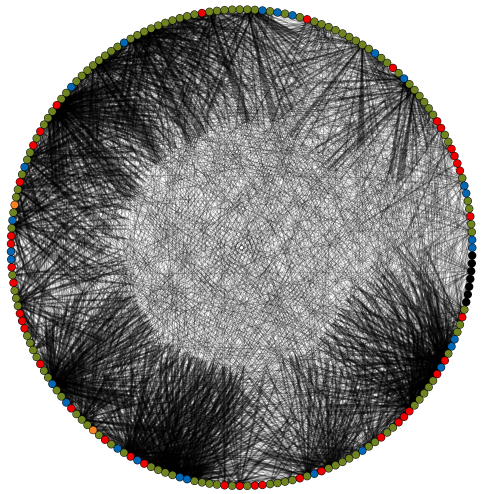


Each circle represents a student and lines connecting students represent romantic relations occuring within the 6 months preceding the interview. Numbers under the figure count the number of times that pattern was observed (i.e. we found 63 pairs unconnected to anyone else).

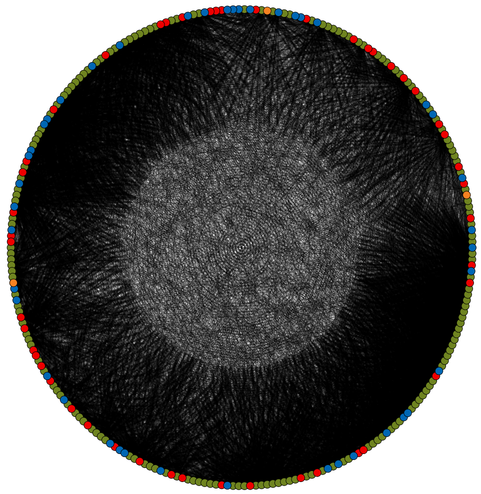
GRAPHS AND THE NETWORKS THEY REPRESENT



UNOS, 2010-12-08



UNOS, 2012-09-10



UNOS, 2014-06-30

NETWORKX

NetworkX is a Python library for storing, manipulating, and analyzing (small- and medium-sized) graphs

- Uses Matplotlib for rendering
- https://networkx.github.io/
- conda install -c anaconda networkx

```
import networkx as nx

G=nx.Graph()
G.add_node("spam”)
G.add_edge(1,2)

print(list(G.nodes()))
print(list(G.edges())) [(1, 2)
```

```
[1, 2, ‘spam’]
[(1,2)]
```

STORING A GRAPH

Three main ways to represent a graph in memory:

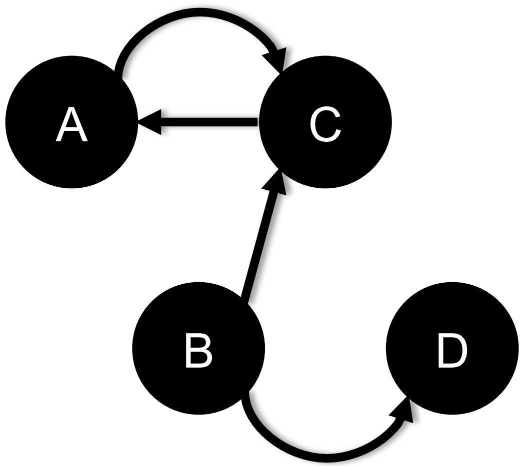
- Adjacency lists
- Adjacency dictionaries
- Adjacency matrix

The storage decision should be made based on the expected use case of your graph:

- Static analysis only?
- Frequent updates to the structure?
- Frequent updates to semantic information?

ADJACENCY LISTS

For each vertex, store an array of the vertices it connects to



Vertex	Neighbors
A	[C]
B	[C, D]
C	[A]
D	[]

Pros: ????????

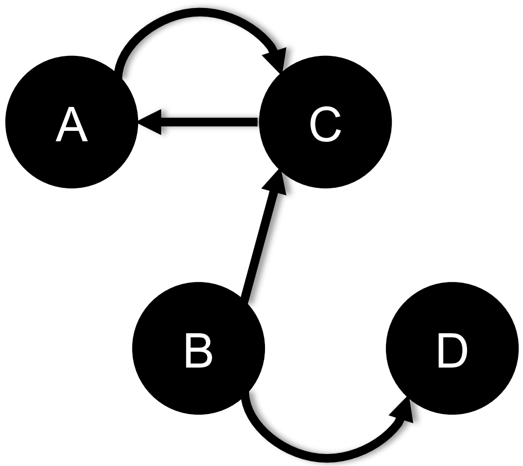
- Iterate over all outgoing edges; easy to add an edge

Cons: ????????

- Checking for the existence of an edge is O(|V|), deleting is hard

ADJACENCY DICTIONARIES

For each vertex, store a dictionary of vertices it connects to



Vertex	Neighbors
A	{C: 1.0}
B	{C: 1.0, D: 1.0}
C	{A: 1.0}
D	{}

Pros: ?????????

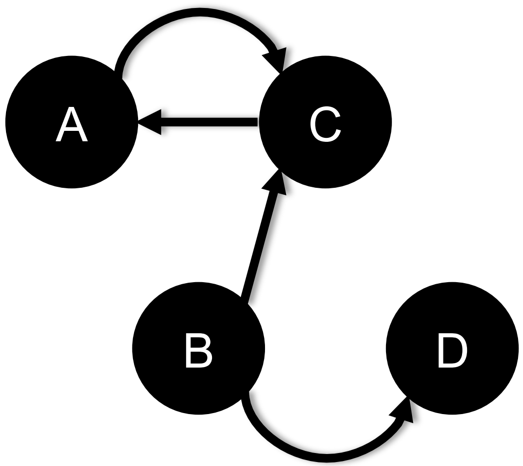
- O(1) to add, remove, query edges

Cons: ?????????

- Overhead (memory, caching, etc)

ADJACENCY MATRIX

Store the connectivity of the graph in a matrix



From

To

	A	B	C	D
A	0	0	1	0
B	0	0	0	0
C	1	1	0	0
D	0	1	0	0

Cons: ?????????

- $O(|V|^2)$ space regardless of the number of edges

Almost always stored as a sparse matrix

NETWORKX STORAGE

NetworkX uses an adjacency dictionary representation

- Built-ins for reading from/to SciPy/NumPy matrices

```
# Make a directed 3-cycle
G=nx.DiGraph()
G.add_edges_from([(‘A’,’B’), (‘B’, ‘C’), (‘C’, ‘A’)])

# Get all out-edges of vertex ’B’
print(G[‘B’])

# Loop over vertices
for v in G.nodes(): print(v)

# Loop over edges
for u,v in G.edges(): print(u, v)
```

ASIDE: GRAPH DATABASES

Traditional relational databases store relations between entities directly in the data (e.g., foreign keys)

- Queries search data, JOIN over relations

Graph databases directly relate data in the storage system using edges (relations) with attached semantic properties

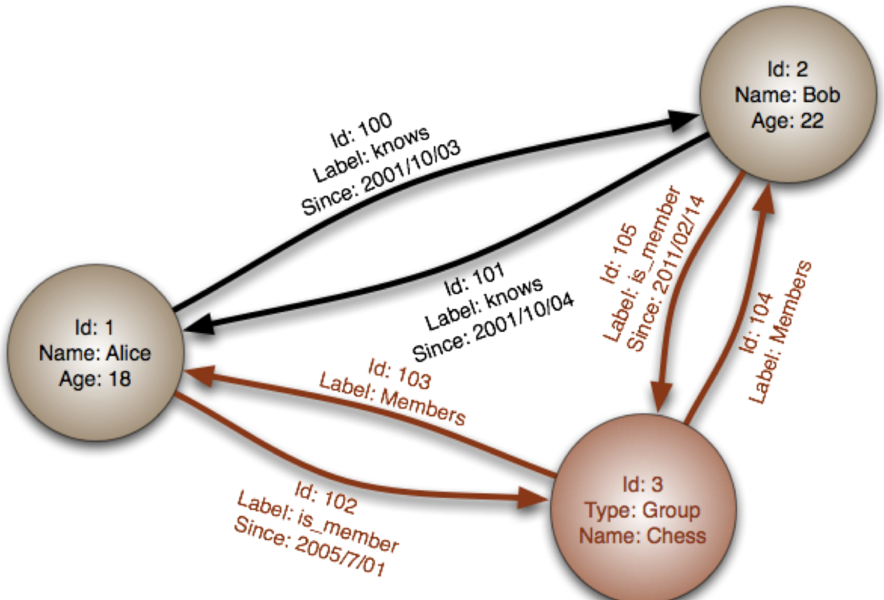
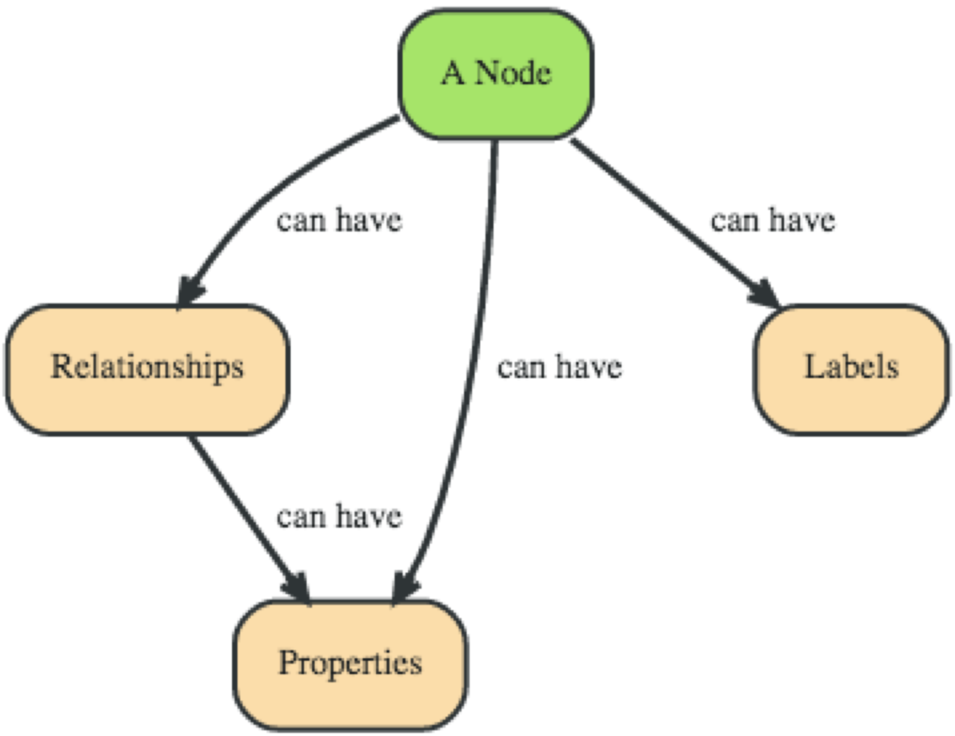


Image thanks to Wikipedia

EXAMPLE GRAPH DATABASE

Two *people*, `John` and `Sally`, are friends.
Both `John` and `Sally` have read the *book*, `Graph Databases`.



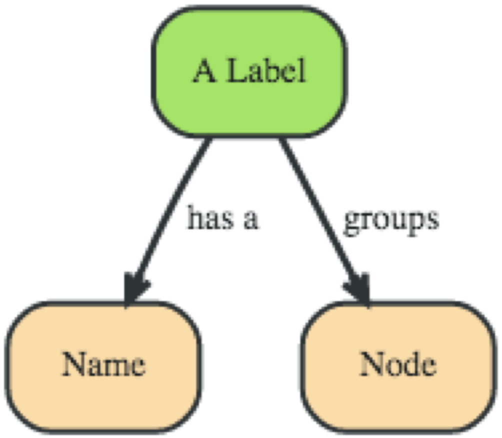
Nodes ??????????

- John
- Sally
- Graph Databases

Thanks to: http://neo4j.com

EXAMPLE GRAPH DATABASE

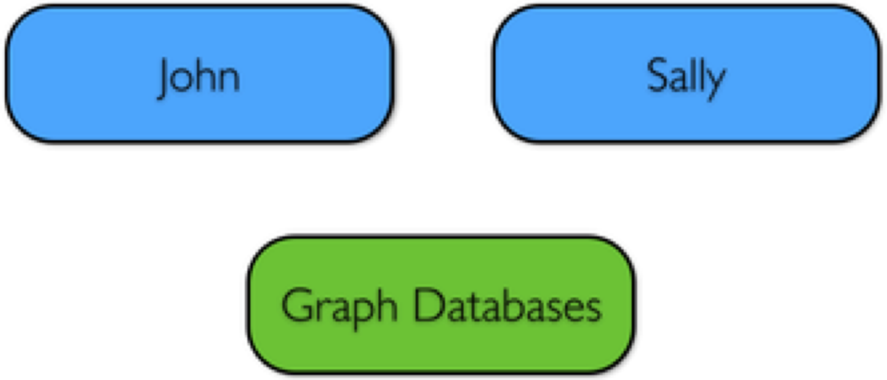
Two *people*, `John` and `Sally`, are friends.
Both `John` and `Sally` have read the *book*, `Graph Databases`.



A named construct that groups nodes into sets

Labels ??????????

- Person
- Book



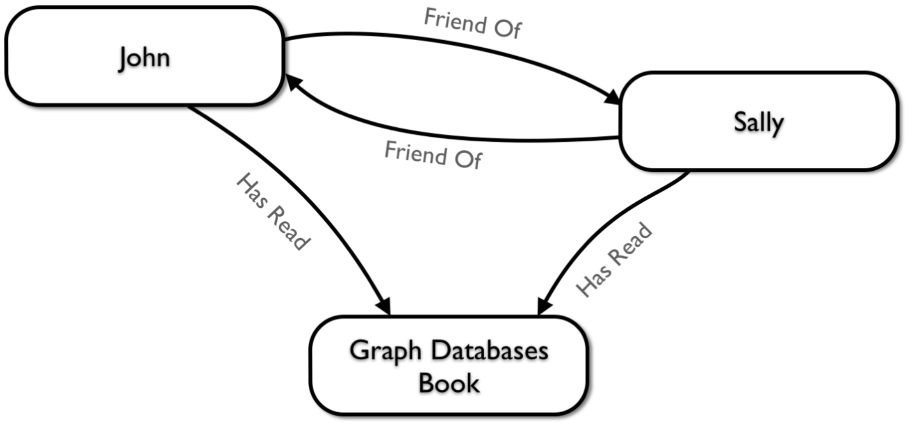
Next: assign labels to the nodes

EXAMPLE GRAPH DATABASE

Two *people*, `John` and `Sally`, are friends.
Both `John` and `Sally` have read the *book*, `Graph Databases`.

Relationships ????????

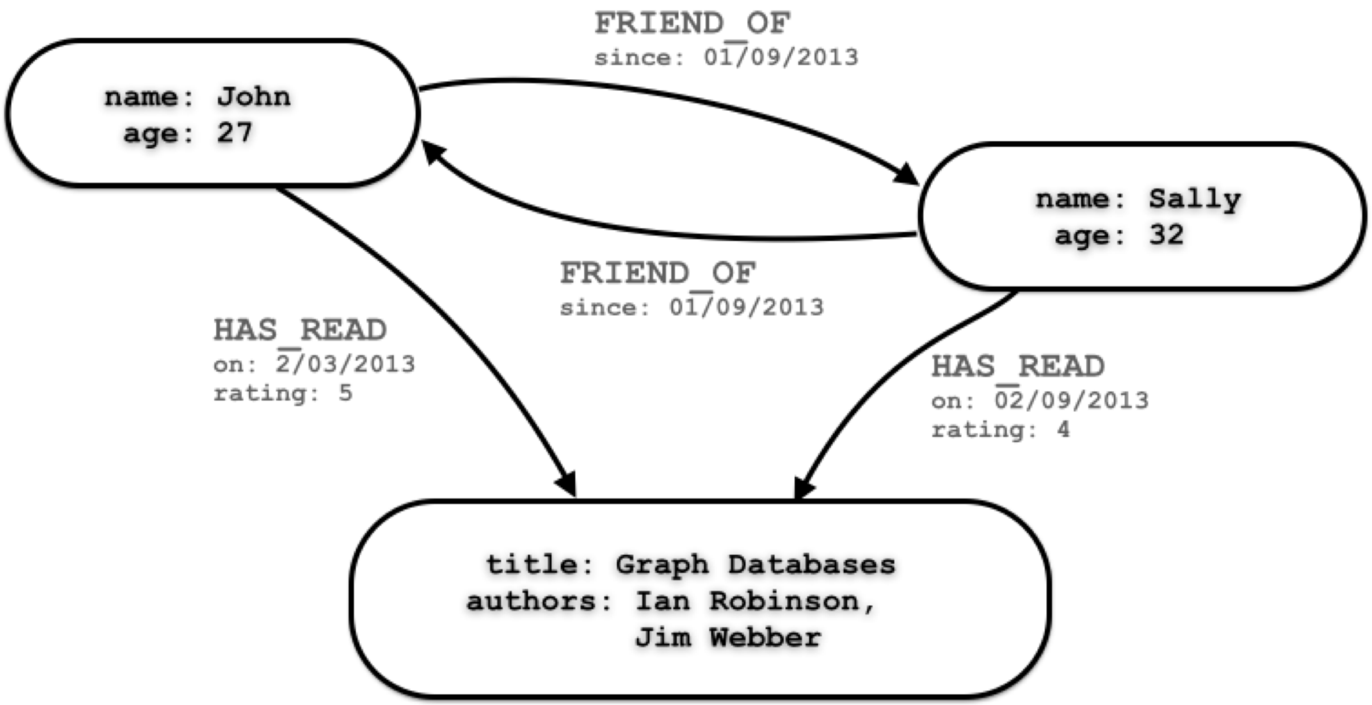
- John is a friend of Sally; Sally is a friend of John
- John has read Graph Databases; Sally has read Graph Databases



EXAMPLE GRAPH DATABASE

Can associate attributes with entities in a key-value way

- Attributes on nodes, relations, labels



EXAMPLE GRAPH DATABASE

Querying graph databases needs a language other than SQL

Recall: graph databases explicitly represent relationships

- Adhere more to an object-oriented paradigm
- May be more suitable for managing ad-hoc data
- May scale better, depending on the query types (no JOINs)

```
# When did Sally and John become friends?
MATCH (sally:Person { name: 'Sally' })
MATCH (john:Person { name: 'John' })
MATCH (sally)-[r:FRIEND_OF]-(john)
RETURN r.since AS friends_since
```

Cypher query

BULBFLOW

Many graph databases out there:

- List found here: https://en.wikipedia.org/wiki/Graph_database

neo4j and Titan are popular, easy-to-use solutions

- https://neo4j.com/
- http://titan.thinkaurelius.com/

Bulbflow is a Python framework that connects to several backing graph-database servers like neo4j

- http://bulbflow.com/
- https://github.com/espeed/bulbs

THE VALUE OF A VERTEX

IMPORTANCE OF VERTICES

Not all vertices are equally important

Centrality Analysis:

- Find out the most important node(s) in one network
- Used as a feature in classification, for visualization, etc …

Commonly-used Measures

- Degree Centrality
- Closeness Centrality
- Betweenness Centrality
- Eigenvector Centrality

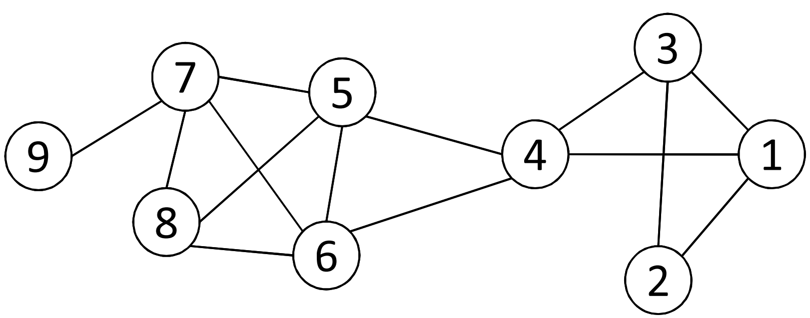
DEGREE CENTRALITY

The importance of a vertex is determined by the number of vertices adjacent to it

- The larger the degree, the more important the vertex is
- Only a small number of vertex have high degrees in many real-life networks

Degree Centrality: $$C_D(v_i) = d_i = \sum_j A_{ij}$$

Normalized Degree Centrality: $$C'_D(v_i) = d_i/(n-1)$$



For vertex 1, degree centrality is 3; Normalized degree centrality is 3/(9-1)=3/8.

CLOSENESS CENTRALITY

"Central" vertices are important, as they can reach the whole network more quickly than non-central vertices

Importance measured by how close a vertex is to other vertices

Average Distance: $$D_{avg}(v_i) = \frac{1}{n-1} \sum_{j \neq i}^{n} g(v_i, v_j)$$

Closeness Centrality:

$$C_C(v_i) = \left[\frac{1}{n-1} \sum_{j \neq i}^{n} g(v_i, v_j) \right]^{-1} = \frac{n-1}{\sum_{j \neq i}^{n} g(v_i, v_j)}$$

CLOSENESS CENTRALITY

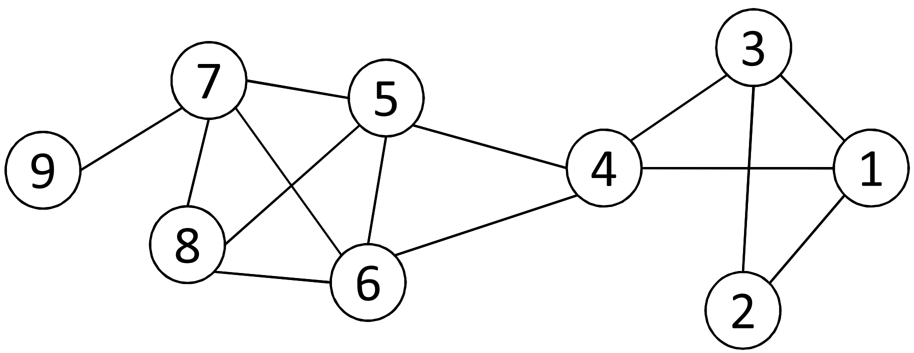


Table 2.1: Pairwise geodesic distance

Node	1	2	3	4	5	6	7	8	9
1	0	1	1	1	2	2	3	3	4
2	1	0	1	2	3	3	4	4	5
3	1	1	0	1	2	2	3	3	4
4	1	2	1	0	1	1	2	2	3
5	2	3	2	1	0	1	1	1	2
6	2	3	2	1	1	0	1	1	2
7	3	4	3	2	1	1	0	1	1
8	3	4	3	2	1	1	1	0	2
9	4	5	4	3	2	2	1	2	0

$$C_C(3) = \frac{9-1}{1+1+1+2+2+3+3+4} = 8/17 = 0.47,$$
$$C_C(4) = \frac{9-1}{1+2+1+1+1+2+2+3} = 8/13 = 0.62.$$

Vertex 4 is more central than vertex 3

BETWEENNESS CENTRALITY

Vertex betweenness **counts the number of shortest paths that pass through one vertex**

Vertices with high betweenness are important in communication and information diffusion

Betweenness Centrality: $$C_B(v_i) = \sum_{v_s \neq v_i \neq v_t \in V, s<t} \frac{\sigma_{st}(v_i)}{\sigma_{st}}$$

σ_{st} : The number of shortest paths between s and t

$\sigma_{st}(v_i)$: The number of shortest paths between s and t that pass v_i

BETWEENNESS CENTRALITY

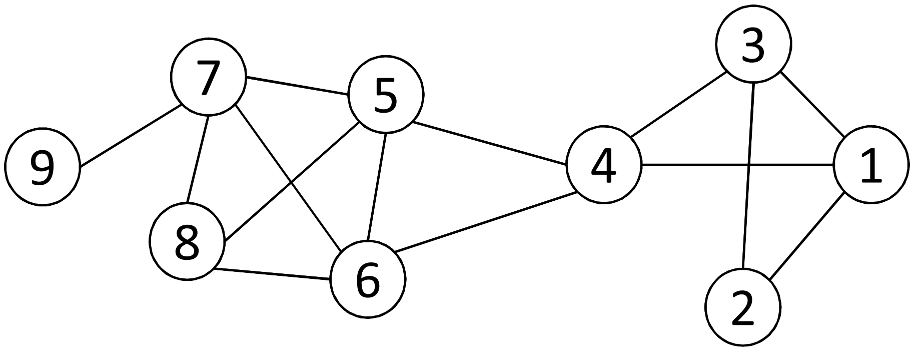


Table 2.2: $\sigma_{st}(4)/\sigma_{st}$

	$s=1$	$s=2$	$s=3$
$t=5$	1/1	2/2	1/1
$t=6$	1/1	2/2	1/1
$t=7$	2/2	4/4	2/2
$t=8$	2/2	4/4	2/2
$t=9$	2/2	4/4	2/2

σ_{st} : The number of shortest paths between s and t

$\sigma_{st}(v_i)$: The number of shortest paths between s and t that pass v_i

$$C_B(v_i) = \sum_{v_s \neq v_i \neq v_t \in V, s<t} \frac{\sigma_{st}(v_i)}{\sigma_{st}}$$

What is the betweenness centrality for node 4 ?????????

EIGENVECTOR CENTRALITY

A vertex's importance is determined by the importance of the friends of that vertex

If one has many important friends, he should be important as well.

$$C_E(v_i) \propto \sum_{v_j \in N_i} A_{ij} C_E(v_j)$$

$$\mathbf{x} \propto A\mathbf{x} \quad \Rightarrow \quad A\mathbf{x} = \lambda \mathbf{x}.$$

The centrality corresponds to the top eigenvector of the adjacency matrix A.

A variant of this eigenvector centrality is the PageRank score.

NETWORKX: CENTRALITY

Many other centrality measures implemented for you!

- https://networkx.github.io/documentation/development/reference/algorithms.centrality.html

Degree, in-degree, out-degree

Closeness

Betweenness

- Applied to both edges and vertices; hard to compute

Load: similar to betweenness

Eigenvector, Katz (provides additional weight to close neighbors)

STRENGTH OF RELATIONSHIPS

WEAK AND STRONG TIES

In practice, connections are not of the same strength

Interpersonal social networks are composed of strong ties (close friends) and weak ties (acquaintances).

Strong ties and weak ties play different roles for community formation and information diffusion

Strength of Weak Ties [Granovetter 1973]

- Occasional encounters with distant acquaintances can provide important information about new opportunities for job search

CONNECTIONS IN SOCIAL MEDIA

Social media allows users to connect to each other more easily than ever.

- One user might have thousands of friends online
- Who are the most important ones among your 300 Facebook friends?

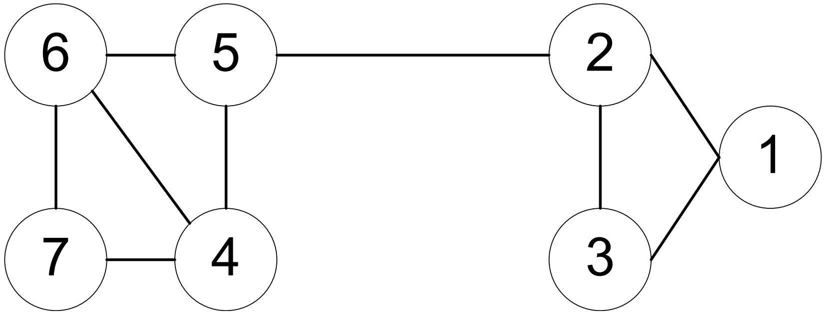
Imperative to estimate the strengths of ties for advanced analysis

- Analyze network topology
- Learn from User Profiles and Attributes
- Learn from User Activities

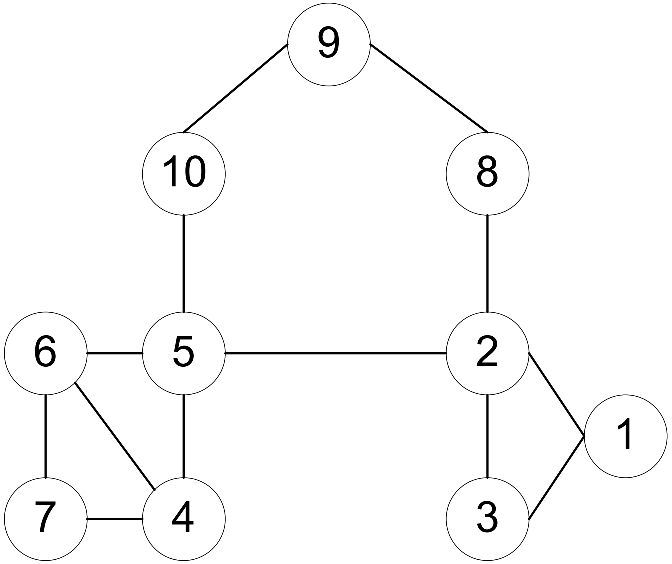
LEARNING FROM NETWORK TOPOLOGY

Bridges connecting two different communities are weak ties

An edge is a bridge if its removal results in disconnection of its terminal vertices



Bridge edge(s) ?????



Bridge edge(s) ?????

"SHORTCUT" BRIDGE

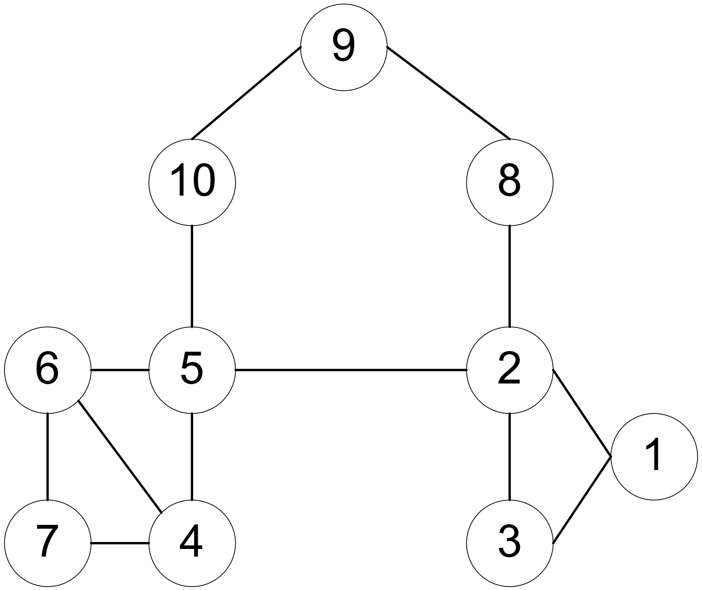
Bridges are rare in real-life networks

Idea: relax the definition by checking if the distance between two terminal vertices increases if the edge is removed

- The larger the distance, the weaker the tie is

Example:

- d(2,5) = 4 if (2,5) is removed
- d(5,6) = 2 if (5,6) is removed
- (5,6) is a stronger tie than (2,5)



NEIGHBORHOOD OVERLAP

Tie strength can be measured based on neighborhood overlap; the larger the overlap, the stronger the tie is.

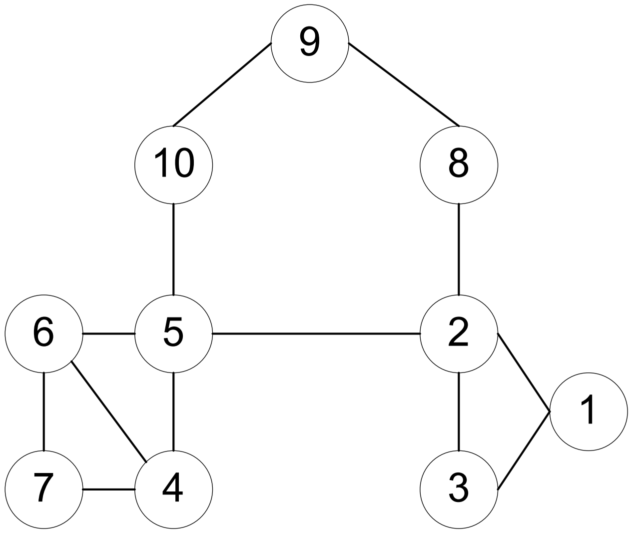
$$overlap(v_i, v_j) = \frac{\text{number of shared friends of both } v_i \text{ and } v_j}{\text{number of friends who are adjacent to at least } v_i \text{ or } v_j} = \frac{|N_i \cap N_j|}{|N_i \cup N_j| - 2}.$$

(-2 in the denominator is to exclude v_i and v_j)

Example:

$$overlap(2, 5) = 0;$$

$$overlap(5, 6) = \frac{|\{4\}|}{|\{2, 4, 5, 6, 7, 10\}| - 2} = 1/4$$



LEARNING FROM PROFILES AND INTERACTIONS

Twitter: one can follow others without followee's confirmation

- The real friendship network is determined by the frequency two users talk to each other, rather than the follower-followee network
- The real friendship network is more influential in driving Twitter usage

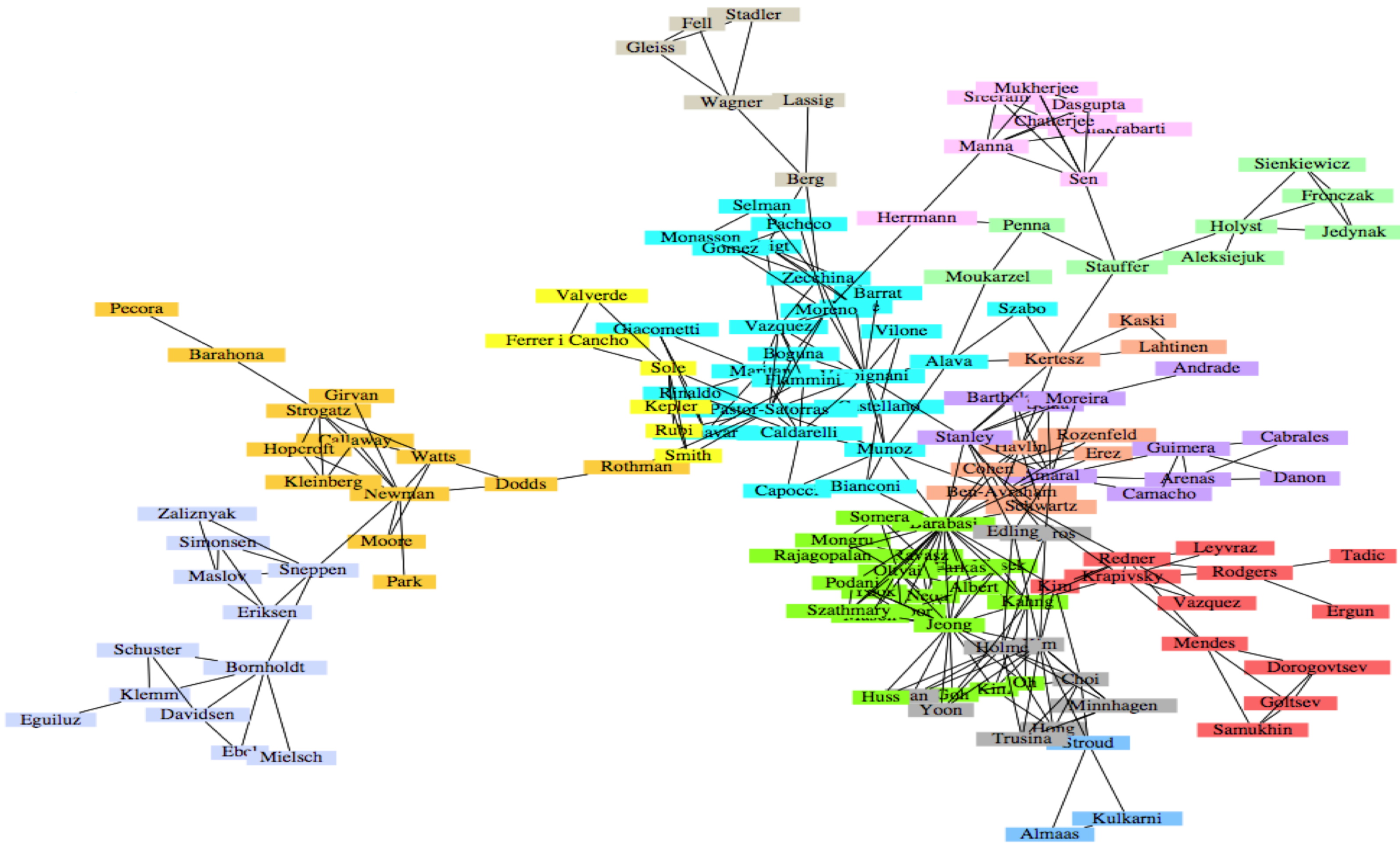
Strengths of ties can be predicted accurately based on various information from Facebook

- Friend-initiated posts, message exchanged in wall post, number of mutual friends, etc.

Learning numeric link strength by maximum likelihood estimation

- User profile similarity determines the strength
- Link strength in turn determines user interaction
- Maximize the likelihood based on observed profiles and interactions

COMMUNITY DETECTION



A co-authorship network of physicists and mathematicians
(Courtesy: Easley & Kleinberg)

WHAT IS A COMMUNITY?

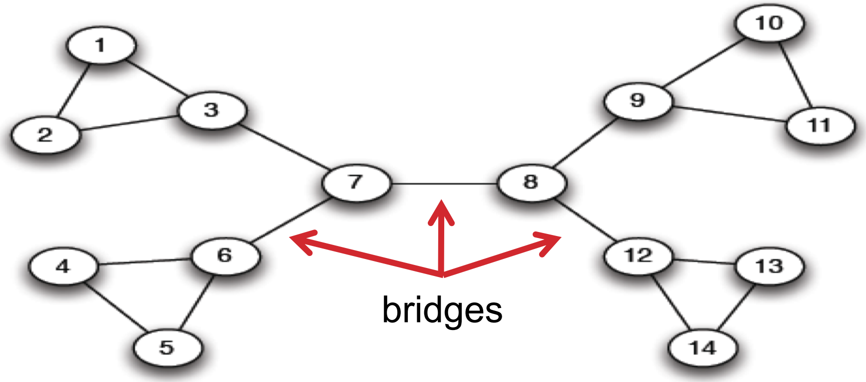
Informally: "tightly-knit region" of the network.

- **How do we identify this region?**
- **How do we separate tightly-knit regions from each other?**

It depends on the definition of tightly knit.

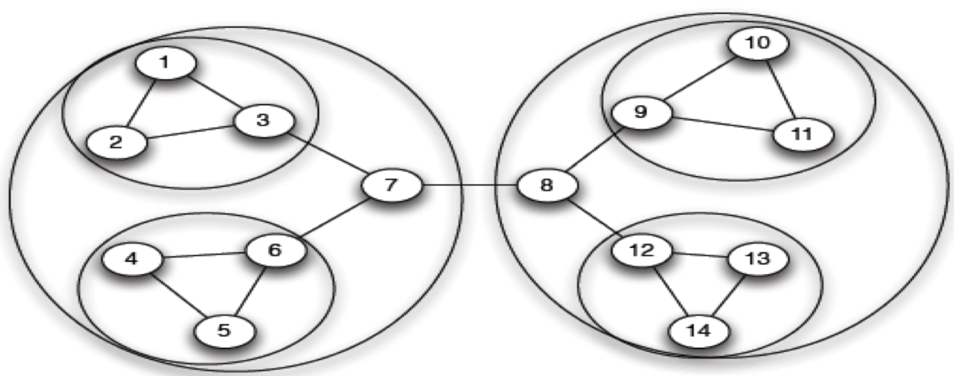
- Regions can be nested
- Examples ?????????
- How do bridges fit into this ?????????

WHAT IS A COMMUNITY?



(a) *A sample network*

Removal of a bridge separates the graph into disjoint components



(b) *Tightly-knit regions and their nested structure*

An example of a nested structure of the communities

(Courtesy: Easley & Kleinberg)

COMMUNITY DETECTION

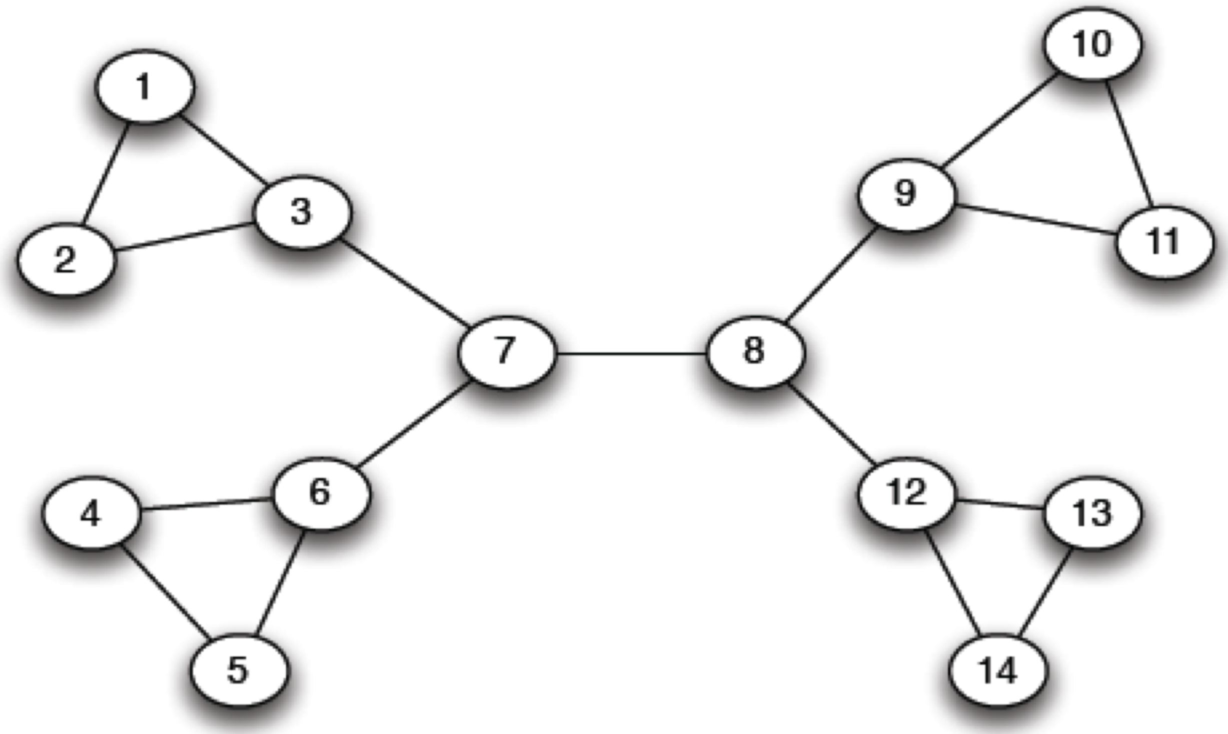
Girvan-Newman Method

- Remove the edges of highest betweenness first.
- Repeat the same step with the remainder graph.
- Continue this until the graph breaks down into individual nodes.

As the graph breaks down into pieces, the tightly knit community structure is exposed.

Results in a hierarchical partitioning of the graph

GIRVAN-NEWMAN METHOD

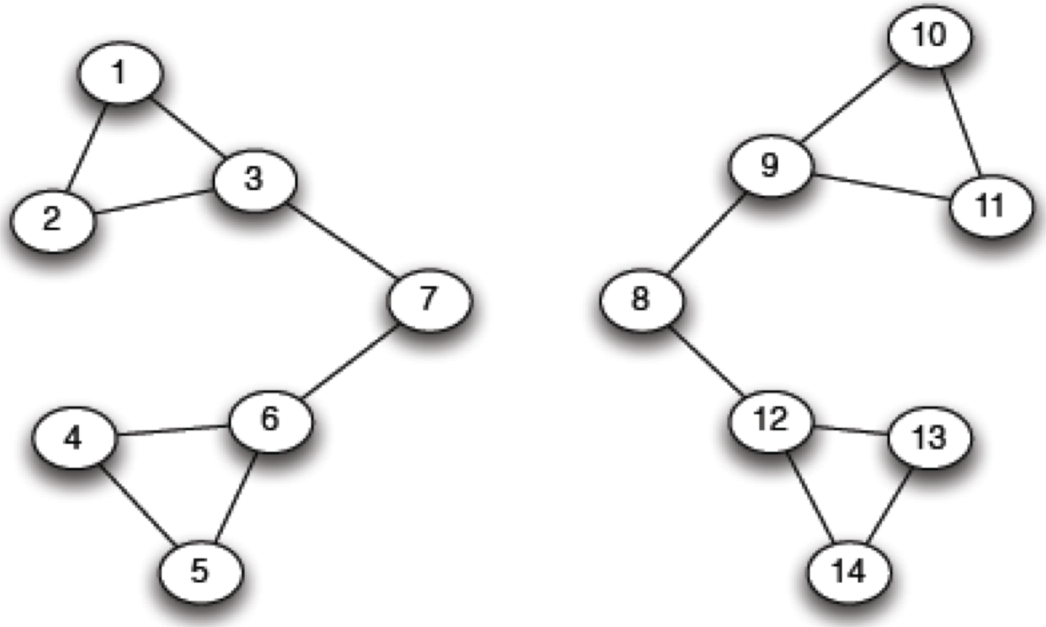


Betweenness(7-8)= 7*7 = 49

Betweenness(1-3) = 1*12 = 12

Betweenness(3-7) = Betweenness(6-7) =
Betweenness(8-9) = Betweenness(8-12) = 3*11= 33

GIRVAN-NEWMAN METHOD

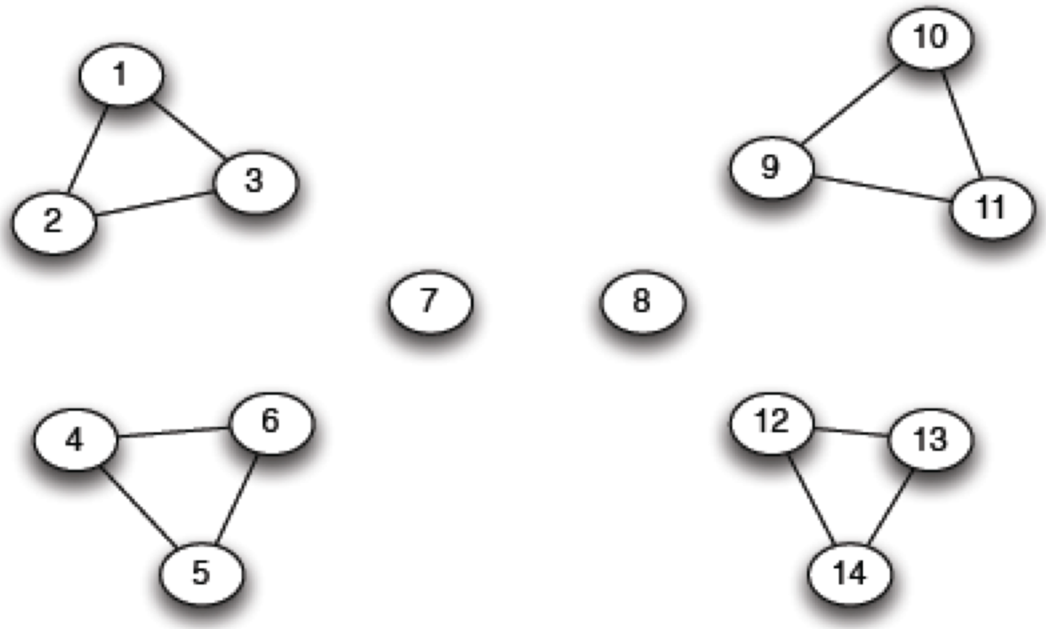


(a) *Step 1*

Betweenness(1-3) = 1*5=5

Betweenness(3-7) = Betweenness(6-7) =
Betweenness(8-9) = Betweenness(8-12) = 3*4 = 12

GIRVAN-NEWMAN METHOD

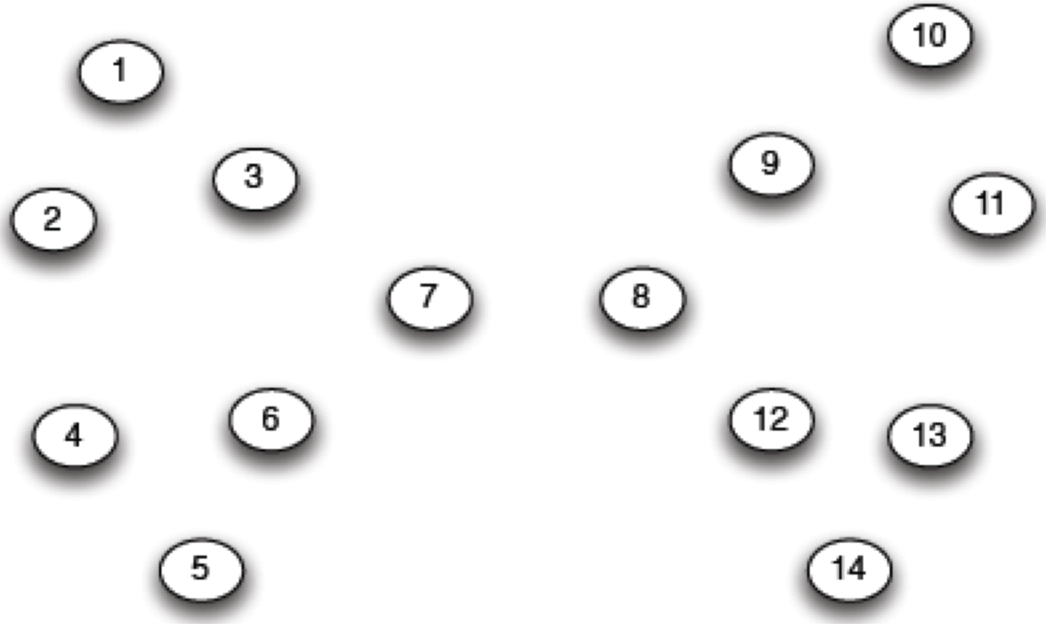


(b) *Step 2*

?????????????????????

Betweenness of every edge = 1

GIRVAN-NEWMAN METHOD




```
G=nx.Graph()

# Returns an iterator over partitions at
# different hierarchy levels
nx.girvan_newman(G)
```

NETWORKX: VIZ

Can render via Matplotlib or GraphViz

```
import matplotlib.pyplot as plt

G=nx.Graph()
nx.draw(G, with_labels=True)

# Save to a PDF
plt.savefig(“my_filename.pdf”)
```

Many different layout engines, aesthetic options, etc

- https://networkx.github.io/documentation/networkx-1.10/reference/drawing.html
- https://networkx.github.io/documentation/development/gallery.html

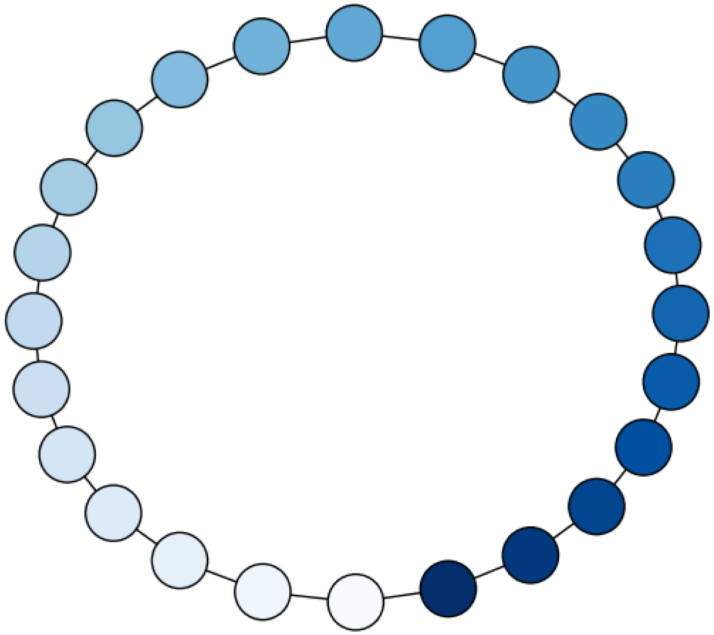
NETWORKX: VIZ

```
# Cycle with 24 vertices
G=nx.cycle_graph(24)

# Compute force-based layout
pos=nx.spring_layout(G,
         iterations=200)

# Draw the graph
nx.draw(G,pos,
      node_color=range(24),
      node_size=800,
      cmap=plt.cm.Blues)

# Save as PNG, then display
plt.savefig("graph.png")
plt.show()
```



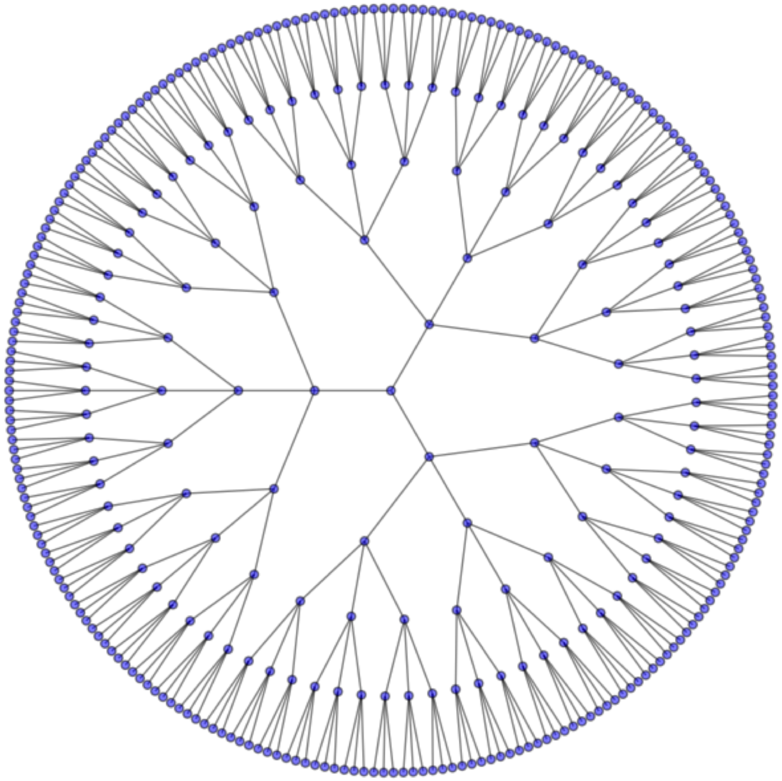
NETWORKX: VIZ

```
# Branch factor 3, depth 5
G = nx.balanced_tree(3, 5)

# Circular layout
pos = graphviz_layout(G,
      prog='twopi', args='')

# Draw 8x8 figure
plt.figure(figsize=(8, 8))
nx.draw(G, pos,
           node_size=20,
           alpha=0.5,
           node_color="blue",
           with_labels=False)

plt.axis('equal')
plt.show()
```



AND NOW:

Words words words!

- Free text and natural language processing in data science
- Bag of words and TF-IDF
- N-Grams and language models
- Sentiment mining

Thanks to: Zico Kolter (CMU) & Marine Carpuat's 723 (UMD)

PRECURSOR TO NATURAL LANGUAGE PROCESSING

For we can easily understand a machine's being constituted so that it can utter words, and even emit some responses to action on it of a corporeal kind, which brings about a change in its organs; for instance, if touched in a particular part it may ask what we wish to say to it; if in another part it may exclaim that it is being hurt, and so on.

(But it never happens that it arranges its speech in various ways, in order to reply appropriately to everything that may be said in its presence, as even the lowest type of man can do.)

-- René Descartes, 1600s

PRECURSOR TO NATURAL LANGUAGE PROCESSING

Turing's Imitation Game [1950]:

- Person A and Person B go into separate rooms
- Guests send questions in, read questions that come out – but they are not told who sent the answers
- Person A (B) wants to convince group that she is Person B (A)

We now ask the question, "What will happen when a machine takes the part of [Person] A in this game?" Will the interrogator decide wrongly as often when the game is played like this as he does when the game is played between [two humans]? These questions replace our original, "Can machines think?"

PRECURSOR TO NATURAL LANGUAGE PROCESSING

Mechanical translation started in the 1930s

- Largely based on dictionary lookups

Georgetown-IBM Experiment:

- Translated 60 Russian sentences to English
- Fairly basic system behind the scenes
- Highly publicized, system ended up spectacularly failing

Funding dried up; not much research in "mechanical translation" until the 1980s …

STATISTICAL NATURAL LANGUAGE PROCESSING

Pre-1980s: primarily based on sets of hand-tuned rules

Post-1980s: introduction of machine learning to NLP

- Initially, decision trees learned what-if rules automatically
- Then, hidden Markov models (HMMs) were used for part of speech (POS) tagging
- Explosion of statistical models for language
- Recent work focuses on purely unsupervised or semi-supervised learning of models

We'll cover some of this in the machine learning lectures!

IBM Watson

NLP IN DATA SCIENCE

In Mini-Project #1, you used `requests` and `BeautifulSoup` to scrape structured data from the web

Lots of data come as unstructured free text: ???????????

- Facebook posts
- Amazon Reviews
- Wikileaks dump

Data science: want to get some meaningful information from unstructured text

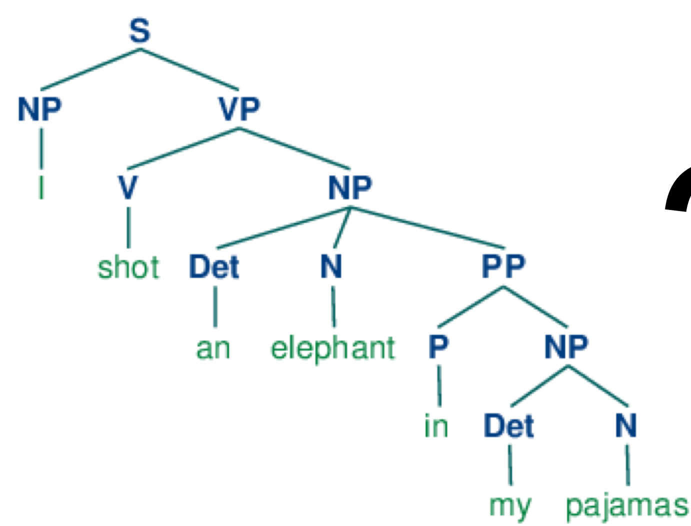
- Need to get some level of understanding what the text says

UNDERSTANDING LANGUAGE IS HARD

One morning I shot an elephant in my pajamas.

How he got into my pajamas, I'll never know.

Groucho Marx



?

UNDERSTANDING LANGUAGE IS HARD

The Winograd Schema Challenge:

- Proposed by Levesque as a complement to the Turing Test

Formally, need to pick out the antecedent of an ambiguous pronoun:

The city councilmen refused the demonstrators a permit because they [feared/advocated] violence.

Terry Winograd

Levesque argues that understanding such sentences requires more than NLP, but also commonsense reasoning and deep contextual reasoning

UNDERSTANDING LANGUAGE IS HARD?

I haven't played it that much yet, but it's shaping to be one of the greatest games ever made! It exudes beauty in every single pixel of it. It's a masterpiece. 10/10

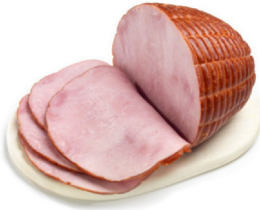
fabchan, March 3, 2017, Metacritic

a horrible stupid game,it's like 5 years ago game,900p 20~30f, i don't play this **** anymore it's like someone give me a **** to play ,no this time sorry,so Nintendo go f yourself pls

Nsucks7752, March 6, 2017, Metacritic

Perhaps we can get some signal (in this case, sentiment) without truly understanding the text …

"SOME SIGNAL"



or

Replication (Part 2 #1) Inbox x

CMSC 320 on Piazza <no-reply@piazza.com> 11:56 PM (1 minute ago) Reply

to me

-- Reply directly to this email above this line to add a comment to the follow up. Or Click here to view.--
A new feedback was posted by Josephine Chow.

does that mean we can use our solution to question 2 to answer question 1? Thank you!

Search or link to this question with @37.

Sign up for more classes at http://piazza.com/umd.

Tell a colleague about Piazza. It's free, after all.

Thanks,
The Piazza Team
--
Contact us at team@piazza.com

You're receiving this email because john@cs.umd.edu is enrolled in CMSC 320 at University of Maryland. Sign in to manage your email preferences or un-enroll from this class.

Possible signals ?????????

≡ SECTIONS The New York Times SUBSCRIBE NOW LOG IN

POLITICS

Trump's New Travel Ban Blocks Migrants From Six Nations, Sparing Iraq

Leer en español

By GLENN THRUSH MARCH 6, 2017 561

President Trump during a meeting in the Roosevelt Room of the White House last week. Al Drago/The New York Times

WASHINGTON — President Trump signed an executive order on Monday blocking citizens of six predominantly Muslim countries from entering the United States, the most significant hardening of immigration policy in generations, even with changes intended to blunt legal and political opposition.

The order was revised to avoid the tumult and protests that engulfed the nation's airports after Mr. Trump signed his first immigration directive on Jan. 27. That order was ultimately blocked by a federal appeals court.

The new order continued to impose a 90-day ban on travelers, but it removed Iraq, a redaction requested by Defense Secretary Jim Mattis, who feared it would hamper coordination to defeat the Islamic State, according to administration officials.

It also exempts permanent residents and current visa holders, and drops language offering preferential status to persecuted religious

"SOME SIGNAL"

What type of article is this?

- Sports
- Political
- Dark comedy

What entities are covered?

- And are they covered with positive or negative sentiment?

Possible signals ????????

ASIDE: TERMINOLOGY

Documents: groups of free text

- Actual documents (NYT article, journal paper)
- Entries in a table

Corpus: a collection of documents

Terms: individual words

- Separated by whitespace or punctuation

TEXT CLASSIFICATION

Is it spam?

Who wrote this paper? (Author identification)

- https://en.wikipedia.org/wiki/The_Federalist_Papers#Authorship
- https://www.uwgb.edu/dutchs/pseudosc/hidncode.htm

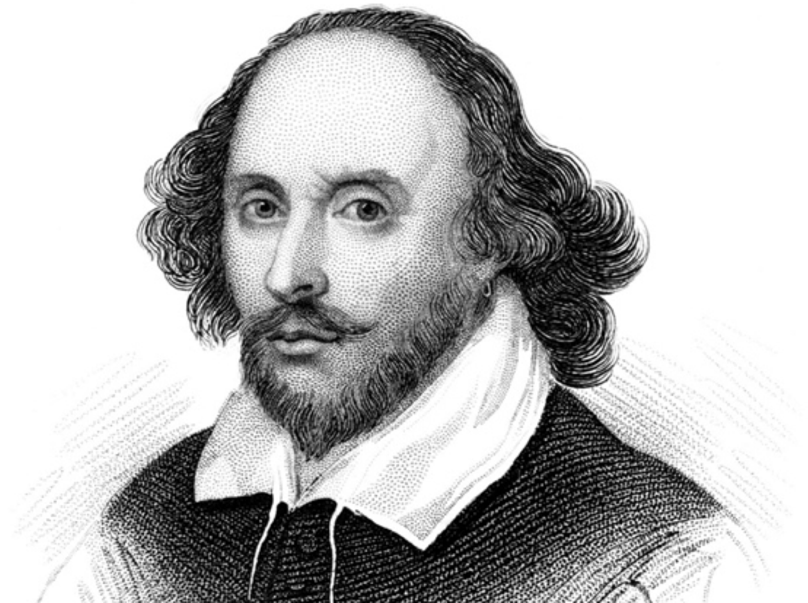
¡Identificación del idioma!

Sentiment analysis

What type of document is this?

When was this document written?

Readability assessment



TEXT CLASSIFICATION

Input:

- A document w
- A set of classes $Y = \{y_1, y_2, \ldots, y_J\}$

Output:

- A predicted class $y \in Y$

(We will spend much more time on classification problems over the next many lectures, this is just a light intro!)

TEXT CLASSIFICATION

Hand-coded rules based on combinations of terms (and possibly other context)

If email *w*:

- Sent from a DNSBL (DNS blacklist) **OR**
- Contains "Nigerian prince" **OR**
- Contains URL with Unicode **OR** …

Then: y_w = spam

Pros: ?????????

- Domain expertise, human-understandable

Cons: ?????????

- Brittle, expensive to maintain, overly conservative

TEXT CLASSIFICATION

Input:

- A document w
- A set of classes $Y = \{y_1, y_2, \ldots, y_J\}$
- A training set of m hand-labeled documents
  $\{(w_1, y_1), (w_2, y_2), \ldots, (w_m, y_m)\}$

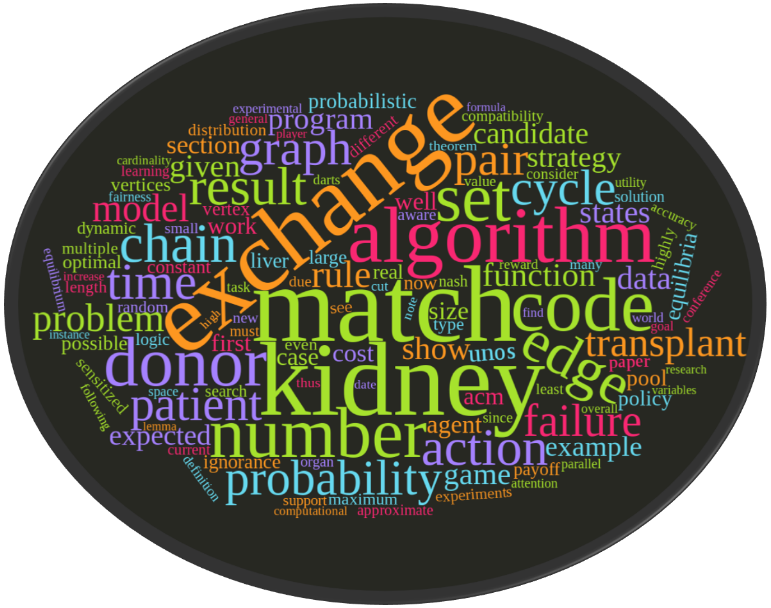
Output:

- A learned classifier $w \rightarrow y$

This is an example of supervised learning

REPRESENTING A DOCUMENT "IN MATH"

Simplest method: bag of words



Represent each document as a vector of word frequencies

- Order of words does not matter, just #occurrences

BAG OF WORDS EXAMPLE

the quick brown fox jumps over the lazy dog

I am he as you are he as you are me

he said the CMSC320 is 189 more CMSCs than the CMSC131

	the	CMSC320	you	he	I	quick	dog	me	CMSCs	…	than
Document 1	2	0	0	0	0	1	1	0	0		0
Document 2	0	0	2	2	1	0	0	1	0	…	0
Document 3	2	1	0	1	0	0	0	0	1		1

TERM FREQUENCY

Term frequency: the number of times a term appears in a specific document

- tf_{ij}: frequency of word *j* in document *i*

This can be the raw count (like in the BOW in the last slide):

- $tf_{ij} \in \{0,1\}$ if word *j* appears or doesn't appear in doc *i*
- $\log(1 + tf_{ij})$ – reduce the effect of outliers
- $tf_{ij} / \max_j tf_{ij}$ – normalize by document i's most frequent word

What can we do with this?

- Use as features to learn a classifier *w* → *y* …!

DEFINING FEATURES FROM TERM FREQUENCY

Suppose we are classifying if a document was written by The Beatles or not (i.e., binary classification):

- Two classes $y \in Y = \{ 0, 1 \} = \{$ not_beatles, beatles $\}$

Let's use $\mathrm{tf}_{ij} \in \{0,1\}$, which gives:

	the	CMSC320	you	he	_	quick	dog	me	CMSCs	…	than
$x_1^T =$	1	0	0	0	0	1	1	0	0		0
$x_2^T =$	0	0	1	1	1	0	0	1	0	…	0
$x_3^T =$	1	1	0	1	0	0	0	0	1		1

$y_1 = 0$

$y_2 = 1$

$y_3 = 0$

Then represent documents with a feature function:

$$\mathbf{f}(\mathbf{x}, y = \text{not\_beatles} = 0) = [\mathbf{x}^T, \mathbf{0}^T, 1]^T$$

$$\mathbf{f}(\mathbf{x}, y = \text{beatles} = 1) = [\mathbf{0}^T, \mathbf{x}^T, 1]^T$$

LINEAR CLASSIFICATION

We can then define weights $\boldsymbol{\theta}$ for each feature

$\boldsymbol{\theta}$ = { <CMSC320, not_beatles> = +1,
<CMSC320, beatles> = -1,
<walrus, not_beatles> = -0.3,
<walrus, beatles> = +1,
<the, not_beatles> = 0,
<the, beatles>, 0, … }

Write weights as vector that aligns with feature mapping

Score ψ of an instance x and class y is the sum of the weights for the features in that class:

$$\psi_{\mathbf{x}y} = \Sigma\, \theta_n\, f_n(\mathbf{x}, y)$$

$$= \boldsymbol{\theta}^{\mathrm{T}}\, \mathbf{f}(\mathbf{x}, y)$$

LINEAR CLASSIFICATION

We have a feature function f(x, *y*) and a score $\psi_{xy} = \theta^\top$ f(x, *y*)

And return the class with highest score!

Compute the score of the document for that class

$$\hat{y} = \arg\max_{y} \theta^\intercal \mathbf{f}(\mathbf{x}, y)$$

For each class y ∈ { not_beatles, beatles }

Where did these weights come from? We'll talk about this in the ML lectures …

(… and also this whole "linear classifier" thing.)

INVERSE DOCUMENT FREQUENCY

Recall:

- tf_{ij}: frequency of word *j* in document *i*

Any issues with this ??????????

- Term frequency gets overloaded by common words

Inverse Document Frequency (IDF): weight individual words negatively by how frequently they appear in the corpus:

$$\text{idf}_j = \log\left(\frac{\#\text{documents}}{\#\text{documents with word } j}\right)$$

IDF is just defined for a word j, not word/document pair j, i

INVERSE DOCUMENT FREQUENCY

	the	CMSC320	you	he	I	quick	dog	me	CMSCs	…	than
Document 1	2	0	0	0	0	1	1	0	0		0
Document 2	0	0	2	2	1	0	0	1	0	…	0
Document 3	2	1	0	1	0	0	0	0	1		1

$$\text{idf}_{\text{the}} = \log\left(\frac{3}{2}\right) = 0.405$$

$$\text{idf}_{\text{you}} = \log\left(\frac{3}{1}\right) = 1.098$$

$$\text{idf}_{\text{CMSC320}} = \log\left(\frac{3}{1}\right) = 1.098$$

$$\text{idf}_{\text{he}} = \log\left(\frac{3}{2}\right) = 0.405$$

TF-IDF

How do we use the IDF weights?

Term frequency inverse document frequency (TF-IDF):

- TF-IDF score: tf_{ij} x idf_j

	the	CMSC320	you	he	I	quick	dog	me	CMSCs	…	than
Document 1	0.8	0	0	0	0	1.1	1.1	0	0		0
Document 2	0	0	2.2	0.8	1.1	0	0	1.1	0	…	0
Document 3	0.8	1.1	0	0.4	0	0	0	0	1.1		1.1

This ends up working better than raw scores for classification and for computing similarity between documents.

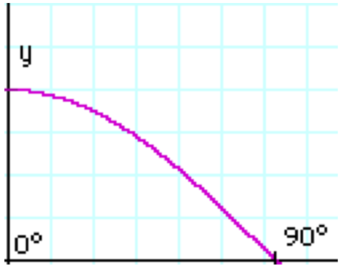
SIMILARITY BETWEEN DOCUMENTS

Given two documents x and y, represented by their TF-IDF vectors (or any vectors), the cosine similarity is:

$$\text{similarity}(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x}^{\mathsf{T}}\mathbf{y}}{|\mathbf{x}| \times |\mathbf{y}|}$$

Formally, it measures the cosine of the angle between two vectors x and y:

- $\cos(0^o) = 1$, $\cos(90^o) = 0$??????????



Similar documents have high cosine similarity; dissimilar documents have low cosine similarity.

NLP IN PYTHON

Natural Language Analyses with NLTK

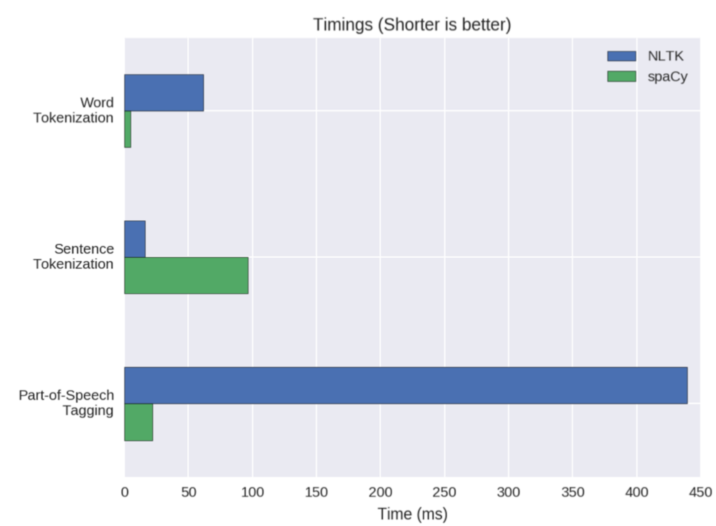
spaCy

Two majors libraries for performing basic NLP in Python:

- Natural Language Toolkit (NLTK): started as research code, now widely used in industry and research
- Spacy: much newer implementation, more streamlined

Pros and cons to both:

- NLTK has more “stuff” implemented, is more customizable
  - This is a blessing and a curse
- Spacy is younger and feature sparse, but can be much faster
- Both are Anaconda packages



NLTK EXAMPLES

```
import nltk

# Tokenize, aka find the terms in, a sentence
sentence = ”A wizard is never late, nor is he early. 
He arrives precisely when he means to.”
tokens = nltk.word_tokenize(sentence)
```

```
LookupError:
**********************************************************************
  Resource 'tokenizers/punkt/PY3/english.pickle' not found.
  Please use the NLTK Downloader to obtain the resource:  >>>
  nltk.download()
  Searched in:
    - '/Users/spook/nltk_data'
    - '/usr/share/nltk_data'
    - '/usr/local/share/nltk_data'
    - '/usr/lib/nltk_data'
    - '/usr/local/lib/nltk_data'
    - ''
***********************************************************
```

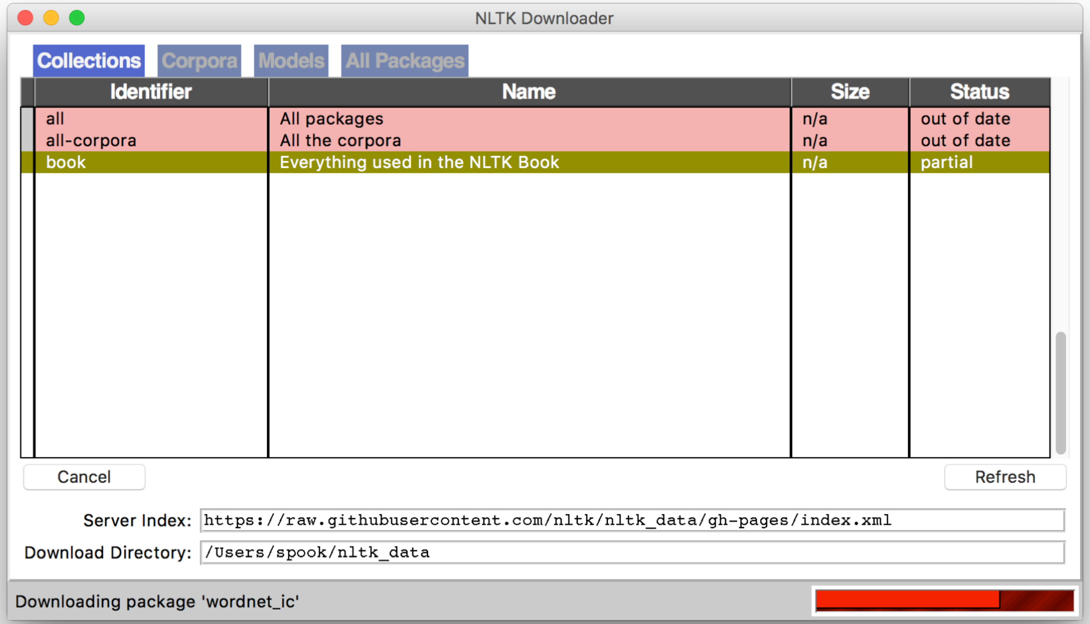
Fool of a Took!

NLTK EXAMPLES

Corpora are, by definition, large bodies of text

- **NLTK relies on a large corpus set to perform various functionalities; you can pick and choose:**

```
# Launch a GUI browser of available corpora
nltk.download()
```




```
# Or download
everything at once!
nltk.download("all")
```

NLTK EXAMPLES

ptb	Penn Treebank	6.1 KB	not installed
punkt	Punkt Tokenizer Models	13.0 MB	installed
qc	Experimental Data for Question Classification	122.5 KB	not installed

```
import nltk

# Tokenize, aka find the terms in, a sentence
sentence = ”A wizard is never late, nor is he early.
He arrives precisely when he means to.”
tokens = nltk.word_tokenize(sentence)
```

```
['A', 'wizard', 'is', 'never', 'late', ',', 'nor',
'is', 'he', 'early', '.', 'He', 'arrives',
'precisely', 'when', 'he', 'means', 'to', '.']
```

(This will also tokenize words like “o’clock” into one term, and “didn’t” into two term, “did” and “n’t”.)

NLTK EXAMPLES

```
# Determine parts of speech (POS) tags
tagged = nltk.pos_tag(tokens)
tagged[:10]
```

```
[('A', 'DT'), ('wizard', 'NN'), ('is', 'VBZ'),
('never', 'RB'), ('late', 'RB'), (',', ','), ('nor',
'CC'), ('is', 'VBZ'), ('he', 'PRP'), ('early', 'RB')]
```

Abbreviation	POS
DT	Determiner
NN	Noun
VBZ	Verb (3rd person singular present)
RB	Adverb
CC	Conjunction
PRP	Personal Pronoun

Full list: https://cs.nyu.edu/grishman/jet/guide/PennPOS.html

NLTK EXAMPLES

```
# Find named entities & visualize
entities = nltk.chunk.ne_chunk( nltk.pos_tag(
nltk.word_tokenize(“””
     The Shire was divided into four quarters, the Farthings already referred
to. North, South, East, and West; and these again each into a number of
folklands, which still bore the names of some of the old leading families,
although by the time of this history these names were no longer found only in
their proper folklands. Nearly all Tooks still lived in the Tookland, but
that was not true of many other families, such as the Bagginses or the
Boffins. Outside the Farthings were the East and West Marches: the Buckland
(see beginning of Chapter V, Book I); and the Westmarch added to the Shire in
S.R. 1462.
    “””)))
entities.draw()
```