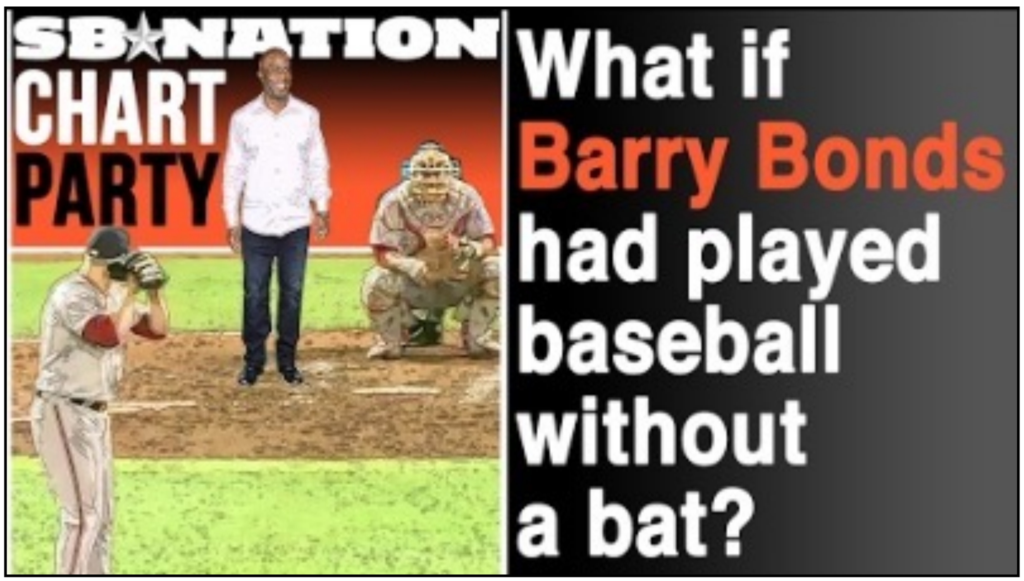
NETWORK ANALYSIS

CS 448B | Fall 2023

MANEESH AGRAWALA

1



2

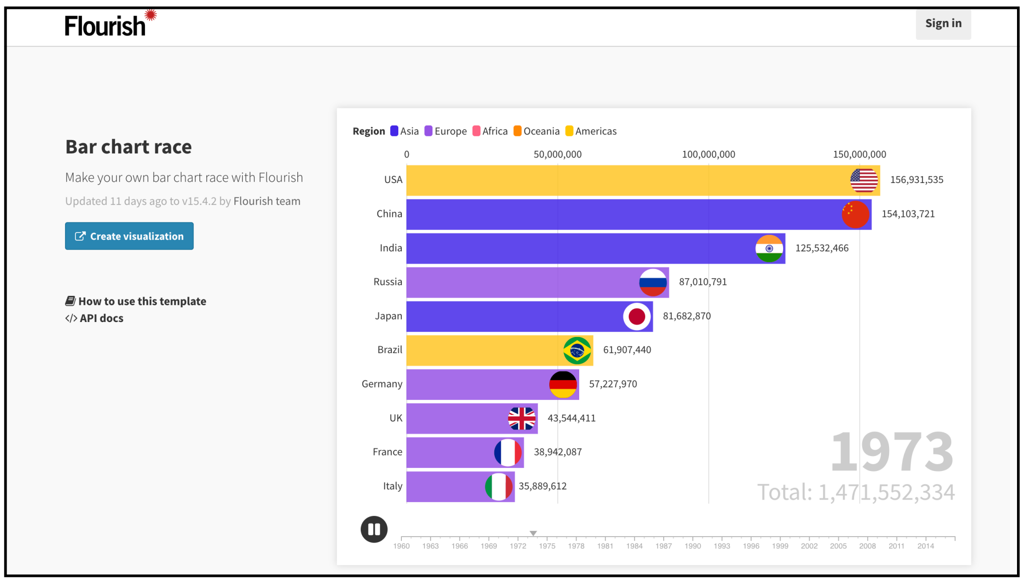
READING RESPONSE: QUESTIONS/THOUGHTS

… When **considering visualizations as a form of art or a medium for storytelling**, **how important is explaining every single detail**? This issue of lack of clarity comes up quite frequently in class and I don't think I fully understand how it is a problem. When you look at a painting, you don't expect to get an explanation for artistic choices like brush strokes. I wonder why when conveying data that those choices seem to matter so much more.

Given the drawbacks of animation that we saw listed in lecture and the advice to use it very carefully because viewers cannot track many moving objects at once, **at what point should we prioritize engagement over clarity in a single viewing?** […] My thought is that a **viewer can replay an animation as many times as they'd like** so that they can focus on different objects of interest, but engagement is something they cannot control.

…I discovered this article on policyviz: https://policyviz.com/2019/08/06/observations-on-animation-in-data-visualization/. **The article raises the point that animation is rarely necessary** and is often used poorly. However, they demonstrate an example of **a very effective use of animation in data visualization: bar chart races**. Bar chart races are popular videos on social media platforms such as Twitter that depict changes in trends over time as a race, using animation to depict both changes in trends, as well as to tell a story over time.

3



4

LAST TIME: NETWORK LAYOUT

5

NODE-LINK GRAPH VISUALIZATION

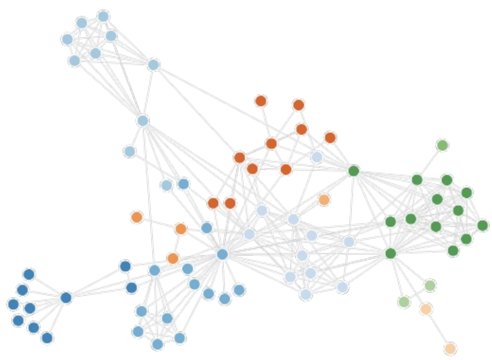
Nodes connected by lines/curves

Sugiyama-Style Layout - arranged by depth
Force-Directed Layout - physical simulation
Attribute-Driven Layout - arranged by value
Constraint-Based Layout – optimization
Arc Diagrams - aligned layout

6

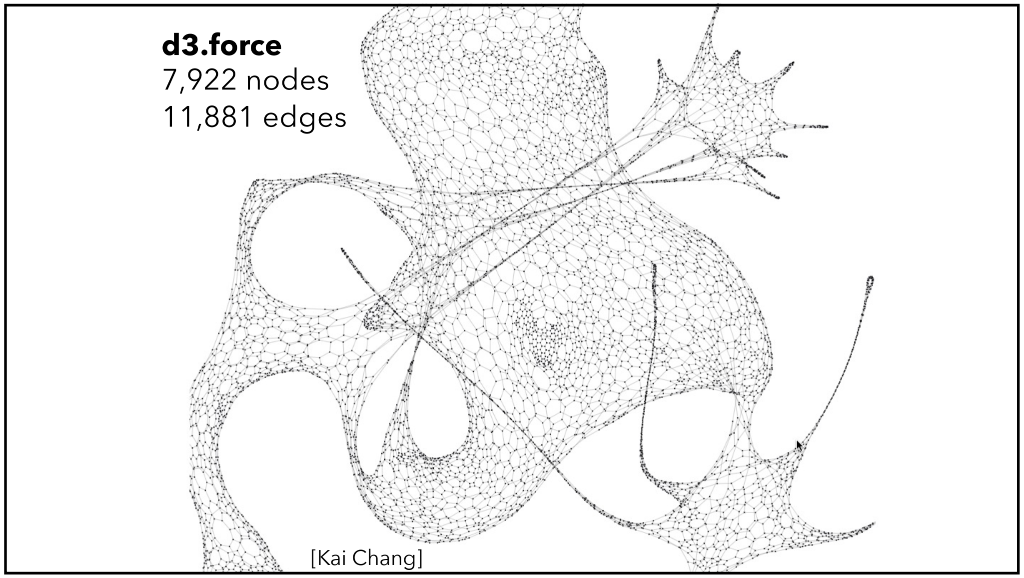
FORCE-DIRECTED LAYOUT

7



Interactive Example: Configurable Force Layout

8



10

Use the Force!

http://mbostock.github.io/d3/talk/20110921/

11

LAYOUT BY PHYSICS SIMULATION

Nodes = charged particles with air resistance — $F = q_i * q_j / d_{ij}^2$, $F = -b * v_i$

Edges = springs — $F = k * (L - d_{ij})$

At each timestep, calculate forces acting on nodes.

Integrate for updated velocities and positions.

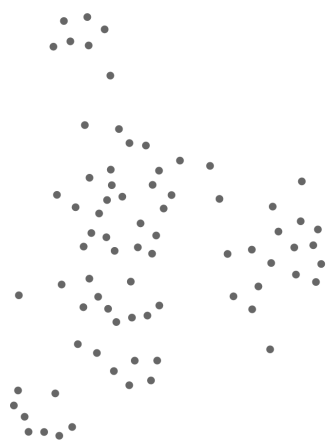
D3's force layout uses **velocity Verlet** integration

Assume uniform mass *m* and timestep *Δt*:

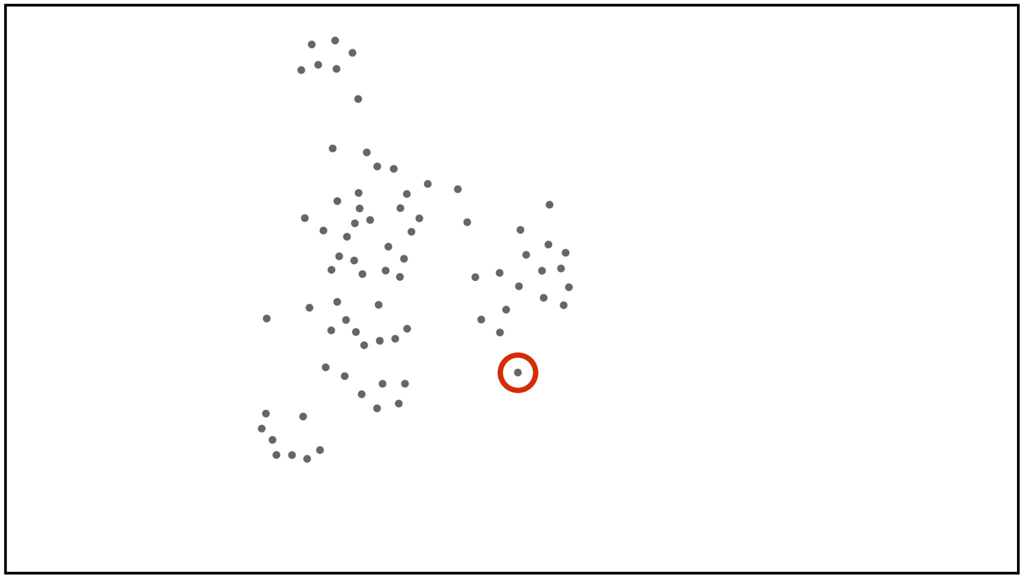
$F = ma \rightarrow F = a \rightarrow F = \Delta v / \Delta t \rightarrow F = \Delta v$

Forces simplify to velocity offsets!

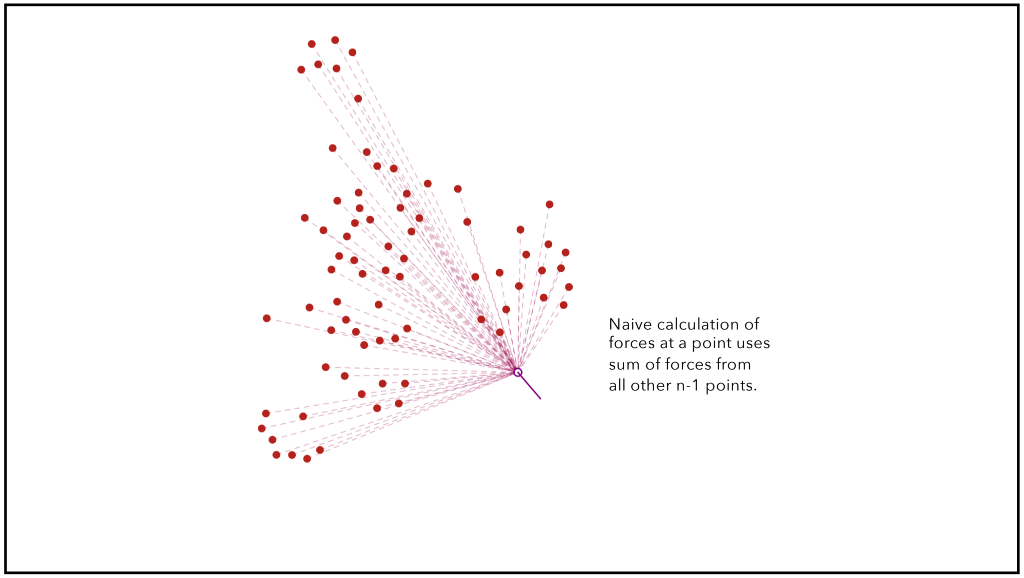
12



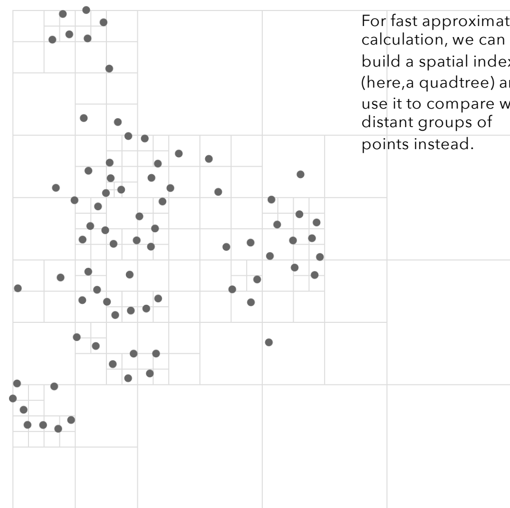
13



14

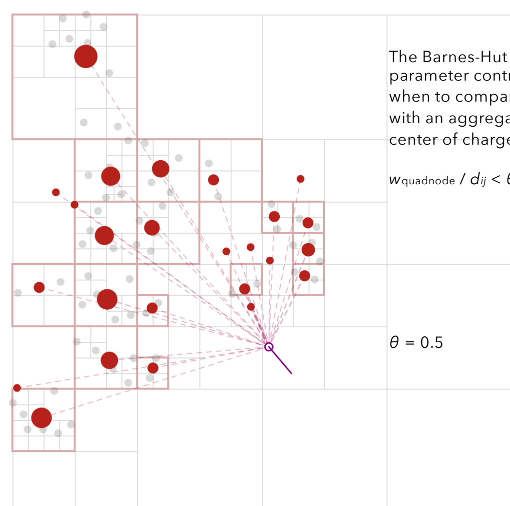

Naive calculation of
forces at a point uses
sum of forces from
all other n-1 points.

15



For fast approximate calculation, we can build a spatial index (here,a quadtree) and use it to compare with distant groups of points instead.

16

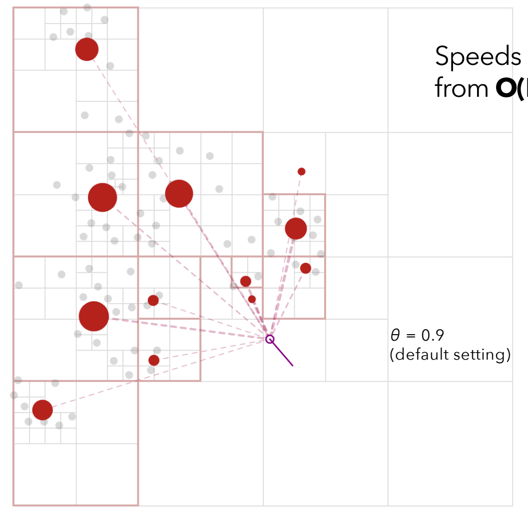


The Barnes-Hut θ parameter controls when to compare with an aggregate center of charge.

$w_{\text{quadnode}} / d_{ij} < \theta$?

$\theta = 0.5$

17



Speeds up algorithm from **O(N²)** to **O(NlogN)**

18

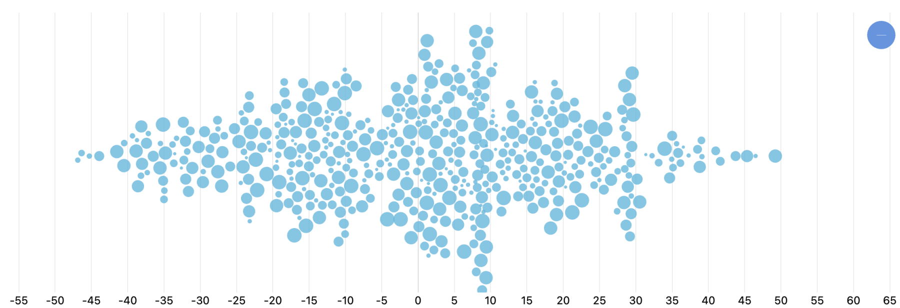
CUSTOMIZED FORCE LAYOUTS

https://www.amcharts.com/demos/beeswarm/

Different forces can be composed to create variety of custom layouts

A **beeswarm plot** can be made by combining:

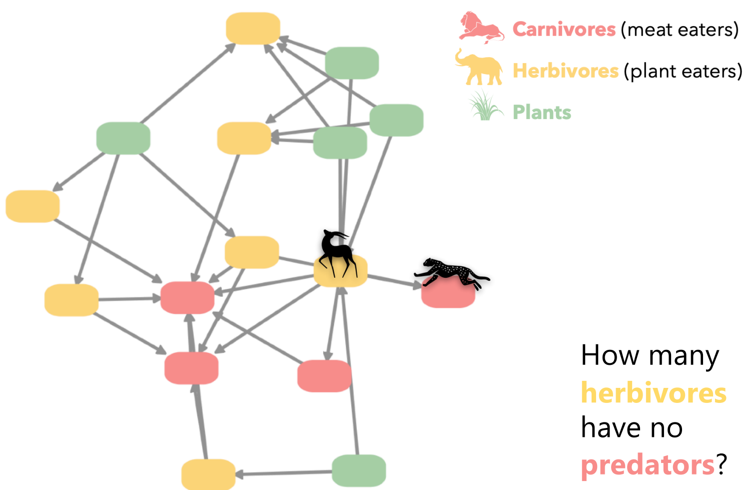
- Attractive **X** and **Y** forces to draw nodes of a certain category to a desired point
- **Collide** force to detect collision & remove overlap



22

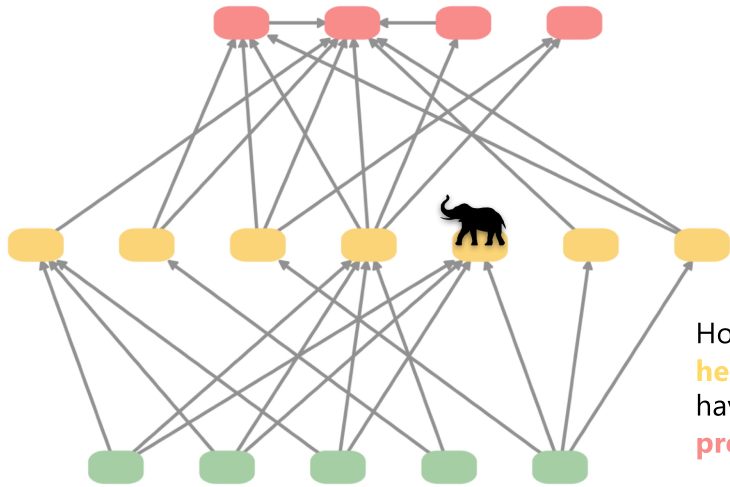
ATTRIBUTE-DRIVEN LAYOUT

23



How many **herbivores** have no **predators**?

24



How many **herbivores** have no **predators**?

25

ATTRIBUTE-DRIVEN LAYOUT

Large node-link diagrams **get messy**!

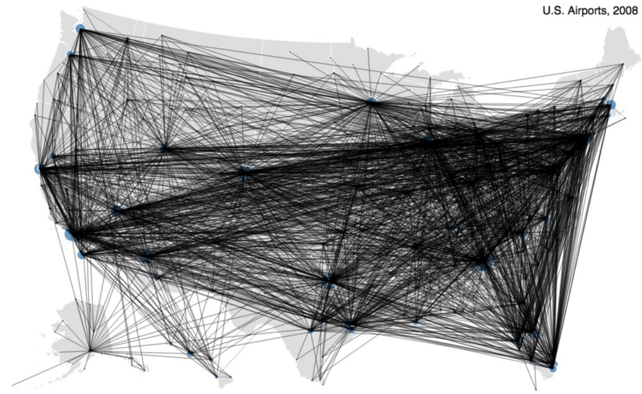
Can we exploit additional structure?

Idea: Use **data fields/attributes** associated with nodes or edges to perform layout (e.g., scatter plot based on node values)

Attributes may also be statistical properties of the graph

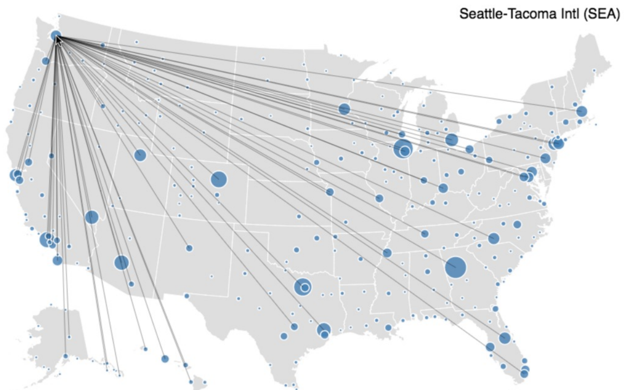
Can apply dynamic queries & brushing on attributes/fields to explore…

26



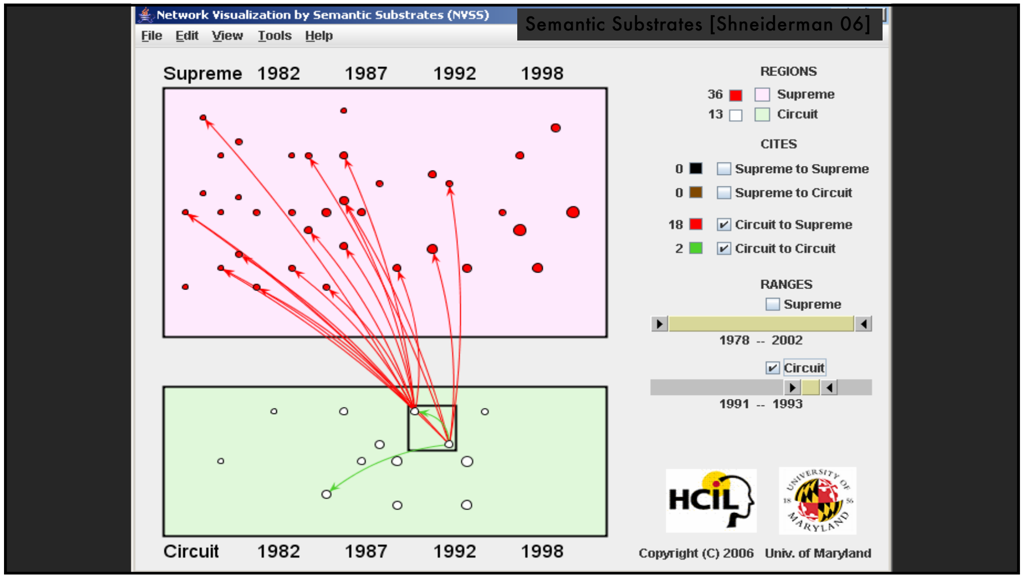
Drawing all edges is not particularly useful here…

28



Node layout determined by geographic location.
Adjacent edges shown on node selection.

29



30

CONSTRAINT-BASED LAYOUT

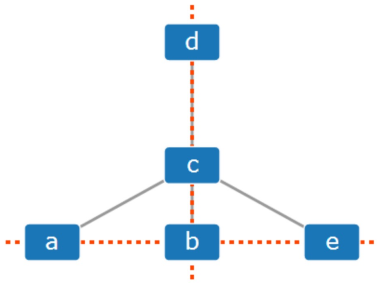
37

CONSTRAINT-BASED LAYOUT

Treat layout as an optimization problem

Define layout using an *energy model* along with *constraint equations* the layout should obey

Use optimization algorithms to solve:



Position Constraints

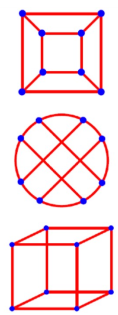
a must be to the ***left*** of b

d, c, and b must have the same ***x position***

a, b, and e must have the same ***y position***

38

OPTIMIZING AESTHETICS



Minimize edge crossings

Minimize area

Minimize line bends

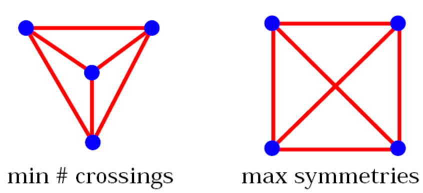
Minimize line slopes

Maximize smallest angle between edges

Maximize symmetry

but, can't do it all

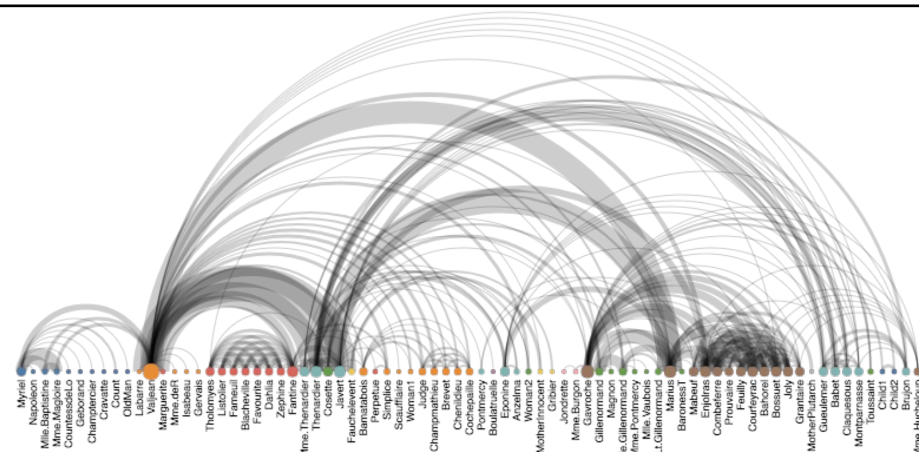
Optimizing these criteria is often NP-Hard, and requires approximations



39

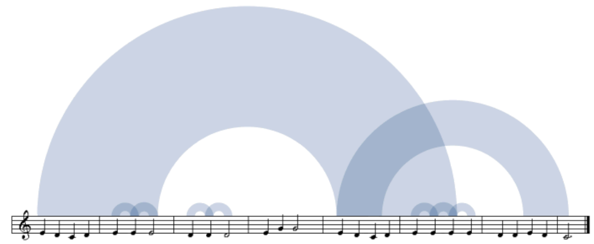
ARC DIAGRAMS

41



Linear node layout, circular arcs show connections.
Layout quality sensitive to node ordering!

42



For example, the picture above was built from the first line of a very simple piece: *Mary Had a Little Lamb*. Each arch connects two identical passages. To clarify the connection between the visualization and the song, in this diagram the score is displayed beneath the arches.

This diagram visualizes the refrain from the folk song *Clementine*. As you would expect, the refrain consists of multiple repetitions of the same passage--and that is exactly what the diagram shows. The score isn't shown in this diagram since the notes would be too small to read.

The Shape of Song
[Wattenberg 2001]

43

NODE-LINK GRAPH VISUALIZATION

Sugiyama-Style Layout - arranged by depth

Force-Directed Layout - physical simulation

Attribute-Driven Layout - arranged by value

Constraint-Based Layout – optimization

Arc Diagrams - aligned layout

44

NODE-LINK GRAPH VISUALIZATION

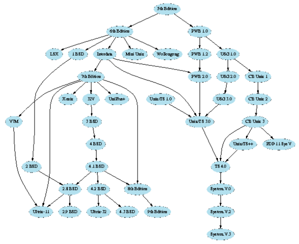
Sugiyama-Style Layout - arranged by depth
- **Good:** Structure-based analysis of hierarchical relationships
- **Bad:** Browsing and path following due to long edges

Force-Directed Layout - physical simulation

Attribute-Driven Layout - arranged by value

Constraint-Based Layout – optimization

Arc Diagrams - aligned layout



45

NODE-LINK GRAPH VISUALIZATION

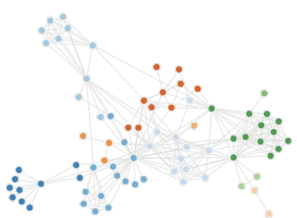
Sugiyama-Style Layout - arranged by depth
- **Good:** Structure-based analysis of hierarchical relationships
- **Bad:** Browsing and path following due to long edges

Force-Directed Layout - physical simulation
- **Good:** Structure-based analysis of closely related elements
- **Bad:** Browsing and summarization of dense networks

Attribute-Driven Layout - arranged by value

Constraint-Based Layout – optimization

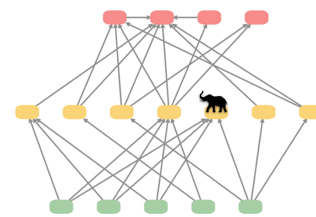
Arc Diagrams - aligned layout



46

NODE-LINK GRAPH VISUALIZATION

Sugiyama-Style Layout - arranged by depth
- **Good:** Structure-based analysis of hierarchical relationships
- **Bad:** Browsing and path following due to long edges



Force-Directed Layout - physical simulation
- **Good:** Structure-based analysis of closely related elements
- **Bad:** Browsing and summarization of dense networks

Attribute-Driven Layout - arranged by value
- **Good:** Enables attribute-based analysis tasks
- **Bad:** Difficult to design layouts appropriate to revealing attributes and network structure

Constraint-Based Layout – optimization

Arc Diagrams - aligned layout

47

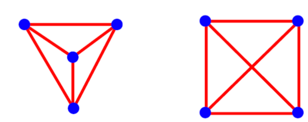
NODE-LINK GRAPH VISUALIZATION

Sugiyama-Style Layout - arranged by depth
- **Good:** Structure-based analysis of hierarchical relationships
- **Bad:** Browsing and path following due to long edges

Force-Directed Layout - physical simulation
- **Good:** Structure-based analysis of closely related elements
- **Bad:** Browsing and summarization of dense networks

Attribute-Driven Layout - arranged by value
- **Good:** Enables attribute-based analysis tasks
- **Bad:** Difficult to design layouts appropriate to revealing attributes and network structure

Constraint-Based Layout – optimization
- **Good:** Graph layout based on structural/aesthetic properties
- **Bad:** Difficult to select appropriate constraints



min # crossings

max symmetries

Arc Diagrams - aligned layout

48

NODE-LINK GRAPH VISUALIZATION

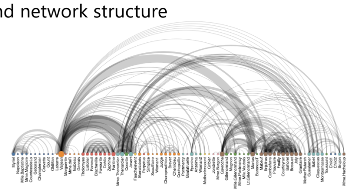
Sugiyama-Style Layout - arranged by depth
- **Good:** Structure-based analysis of hierarchical relationships
- **Bad:** Browsing and path following due to long edges

Force-Directed Layout - physical simulation
- **Good:** Structure-based analysis of closely related elements
- **Bad:** Browsing and summarization of dense networks

Attribute-Driven Layout - arranged by value
- **Good:** Enables attribute-based analysis tasks
- **Bad:** Difficult to design layouts appropriate to revealing attributes and network structure

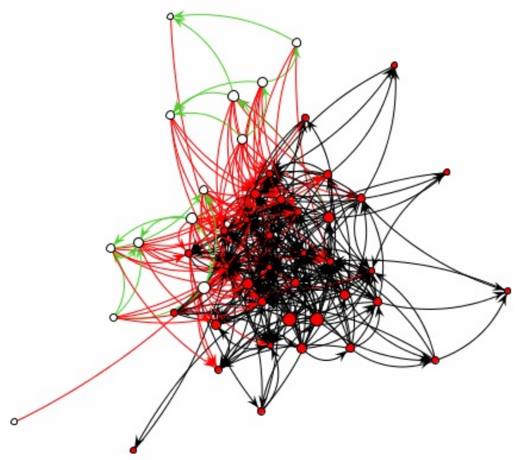
Constraint-Based Layout – optimization
- **Good:** Graph layout based on structural/aesthetic properties
- **Bad:** Difficult to select appropriate constraints

Arc Diagrams - aligned layout
- **Good:** Summarization and comparison of overall structure
- **Bad:** Order matters for node layout; Structure-based and path following



49

LIMITATIONS OF NODE-LINK LAYOUTS



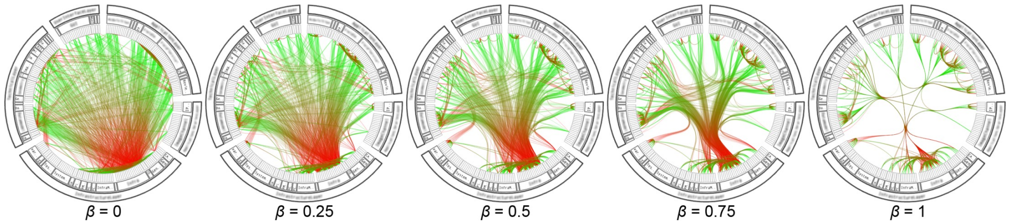
Edge crossings and occlusions!
Poor scalability…

50

HIERARCHICAL EDGE BUNDLING

51

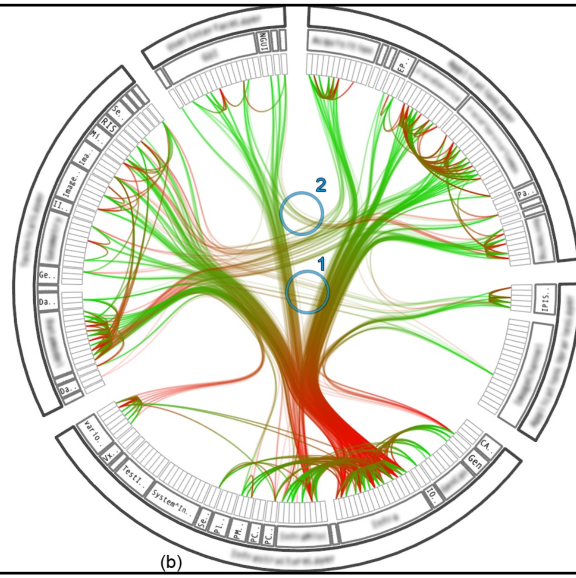
HIERARCHICAL EDGE BUNDLING



Given a tree with additional *adjacency* edges (usually between leaves)

Bundle edges with varying amounts of tension – helping to reveal common connections between subtrees

53

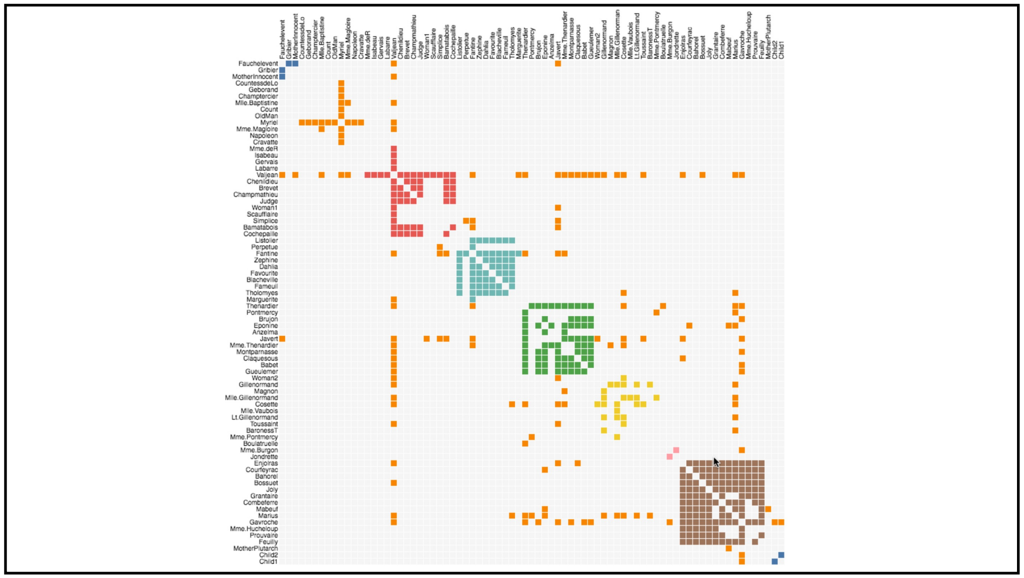


(b)

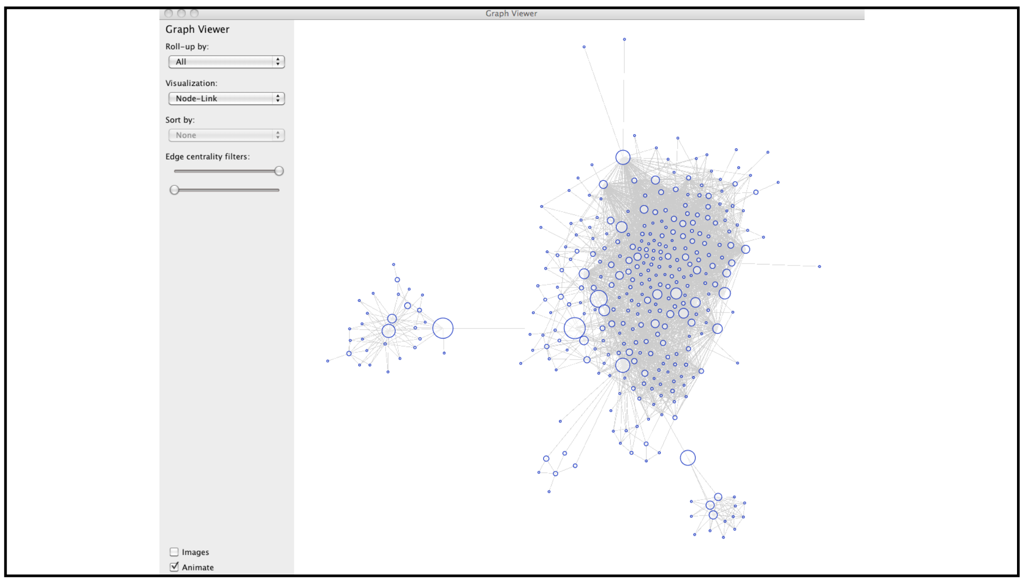
54

MATRIX DIAGRAMS

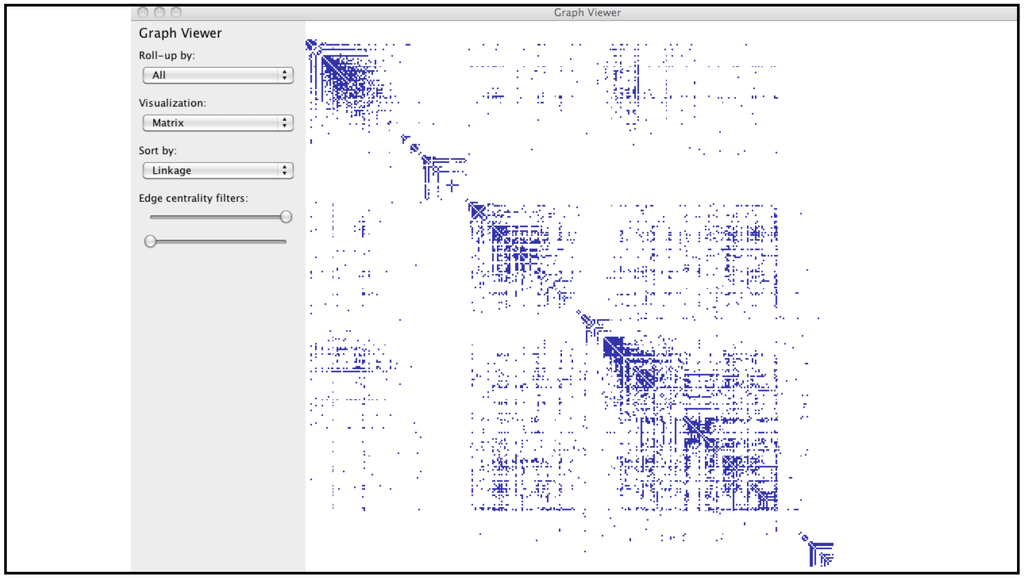
59



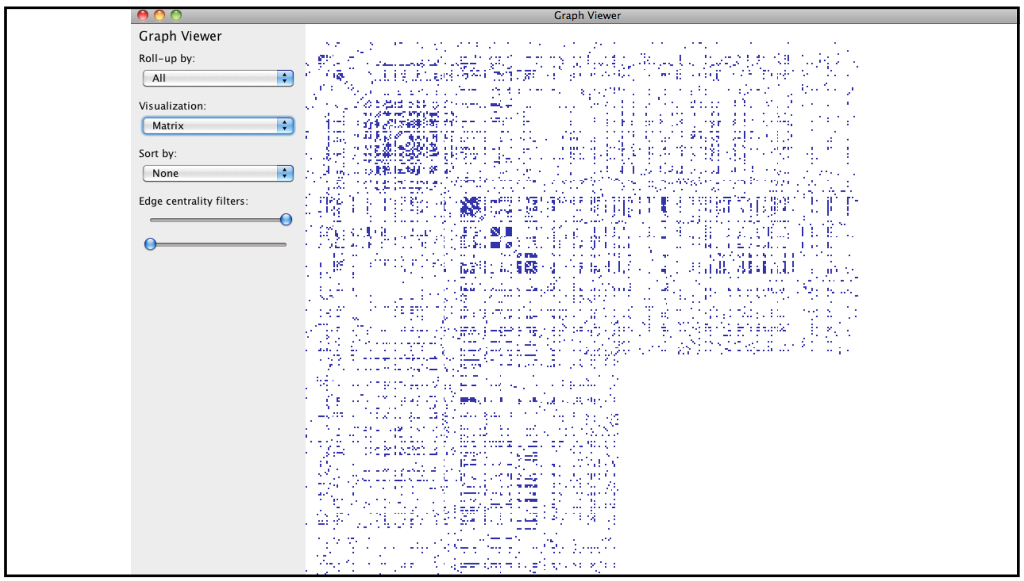
60


Graph Viewer
Graph Viewer
Roll-up by:
All
Visualization:
Node-Link
Sort by:
None
Edge centrality filters:
Images
Animate

61


Graph Viewer
Graph Viewer
Roll-up by:
All
Visualization:
Matrix
Sort by:
Linkage
Edge centrality filters:

62


Graph Viewer
Graph Viewer
Roll-up by:
All
Visualization:
Matrix
Sort by:
None
Edge centrality filters:

63

SUMMARY: TREES AND NETWORKS

Tree Layout

Indented / Node-Link / Enclosure / Layers

Focus+Context techniques for scale

Graph Layout

Sugiyama Layout

Force-Directed Layout

Attribute-Driven Layout

Constraint Layout

Arc Diagrams

Matrix Diagrams

64

ANNOUNCEMENTS

65

FINAL PROJECT

Design Review Nov 27 and 29

Data analysis/explainer

Analyze dataset in depth & make a visual explainer

Deliverables

An article with multiple different interactive visualizations
Short video (2 min) demoing and explaining the project

Schedule

Project proposal: Mon 11/6
Design Review and Feedback: 9th week of quarter, 11/27 and 11/29
Final code and video: Sun 12/10 8pm

Grading

Groups of up to 3 people, graded individually
Clearly report responsibilities of each member

66

FINAL PROJECT GUIDELINES

Consider the audience

Your visual explainer should be of interest to a group of people beyond your immediate circle (an explainer about your own Spotify data unlikely be of interest to others you don't know)

Pick relatively less explored topics/datasets

Do some research on what has already been done for the topic/dataset(s)

Certain data like songs (e.g. Spotify) or movies (e.g. IMDB) are already well analyzed and should be avoided, unless you want to try to take a very different angle or use innovative analysis methods

Develop a narrative

In the early stages of the analysis process, try to uncover patterns to help you form and shape a narrative through-line for the explainer

67

FINAL PROJECT GUIDELINES

Design visualization interactions

Choose base visualizations that can support a high level of interactivity

- Bubble charts, tree maps, and word clouds typically aren't the most effective choices

Design interactive features that would enable viewers to interact with the data in a way that strengthens your narrative

- Tooltip is typically not enough interaction

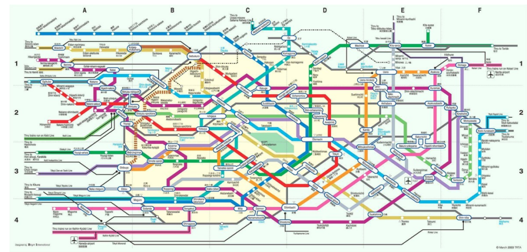
Draw inspiration from sites like the New York Times and the Pudding

68

NETWORK ANALYSIS

69

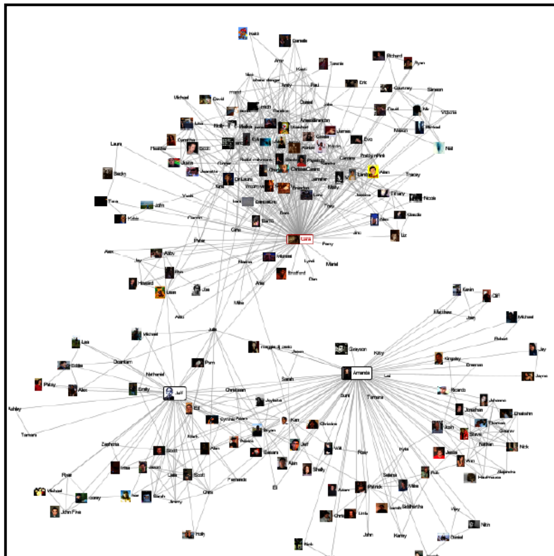
TRANSPORTATION



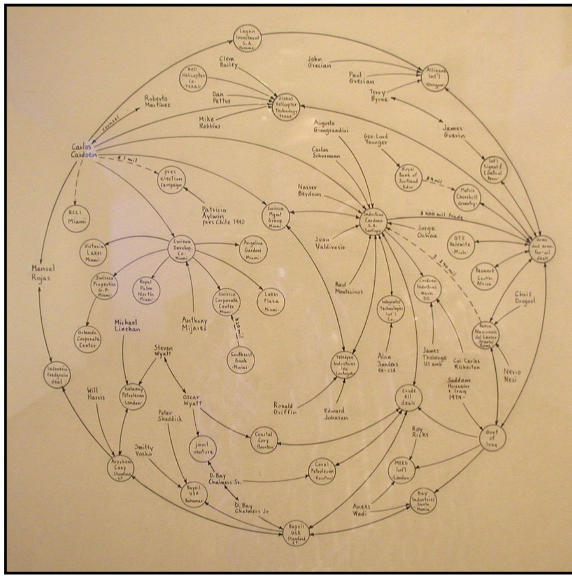
http://www.lx97.com/maps/

70

SOCIAL NETWORKS



72



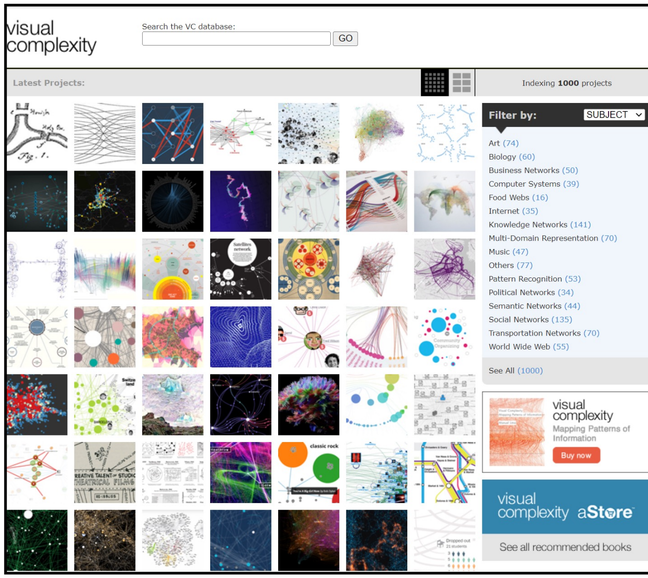
"Mark Lombardi (1951-2000) draws on the major political and financial scandals of the day to create large-scale linear diagrams that at first glance look like celestial maps; a closer reading reveals the intricate web of connections that lurk beneath current headlines.

Lombardi, M. 'George W. Bush, Harken Energy and Jackson Stephens, ca 1979–90'

73

ACTORS & MOVIES (BIPARTITE)

74

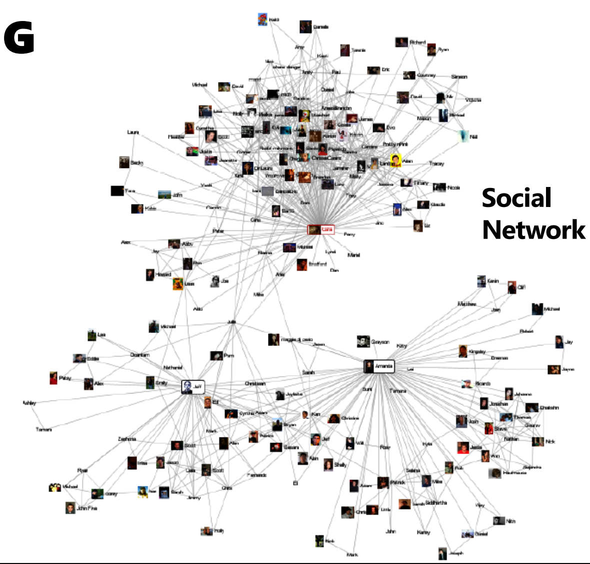


http://www.visualcomplexity.com/vc/

75

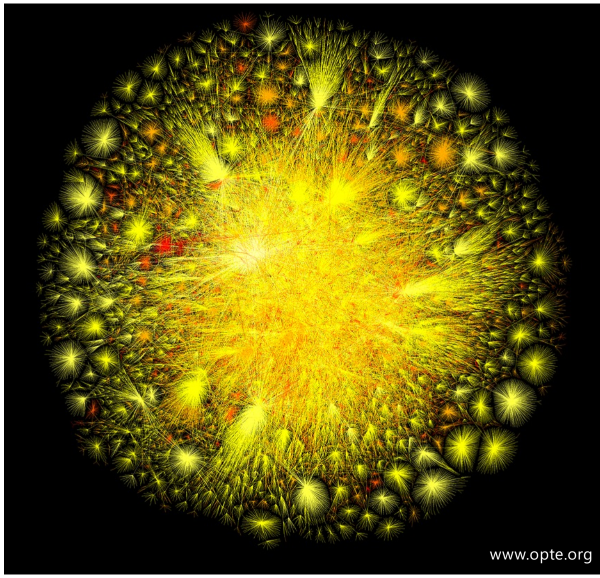
CHARACTERIZING NETWORKS

What does it look like?



Social Network

77



Size?
Density?
Centrality?
Clustering?
Components?
Cliques?
Motifs?
Avg. path length?
…

78

TODAY

Learning Objectives

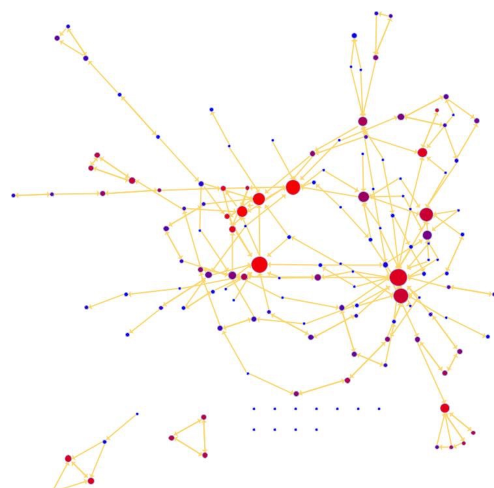
1. Measures of importance/centrality
2. Extracting community structure
3. Simulating network models

79

CENTRALITY

80

HOW FAR APART ARE THINGS?



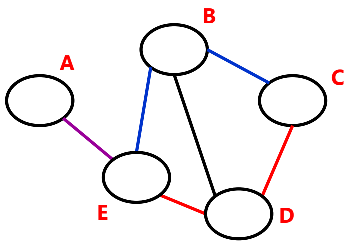
81

DISTANCE: SHORTEST PATHS

Shortest path (geodesic path)

The shortest sequence of links connecting two nodes
Not always unique

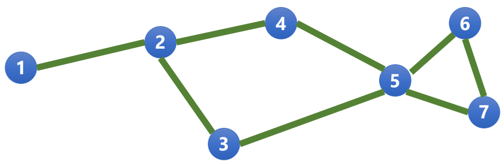
A and C are connected by 2 shortest paths
A – E – B - C
A – E – D - C



82

DISTANCE: SHORTEST PATHS

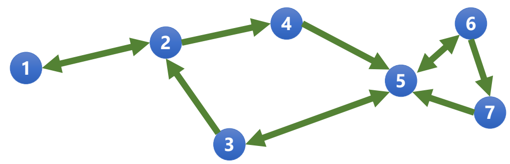
Shortest path from 2 to 3: 1



83

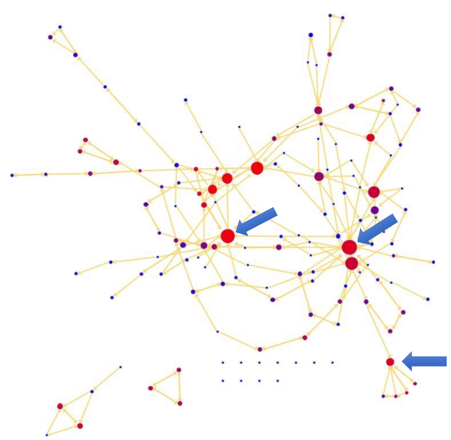
DISTANCE: SHORTEST PATHS

Shortest path from 2 to 3?



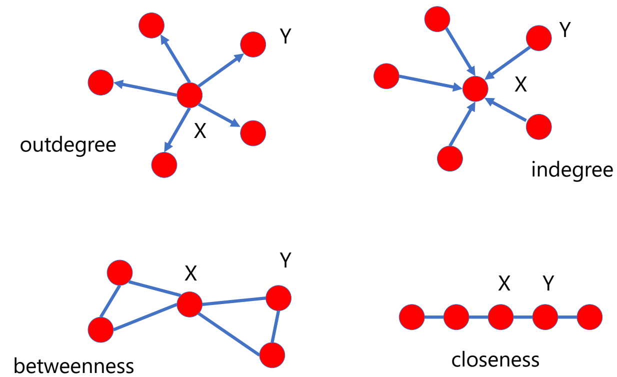
84

MOST IMPORTANT NODE



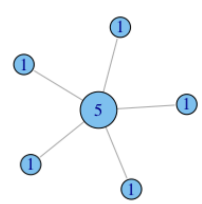
85

CENTRALITY



86

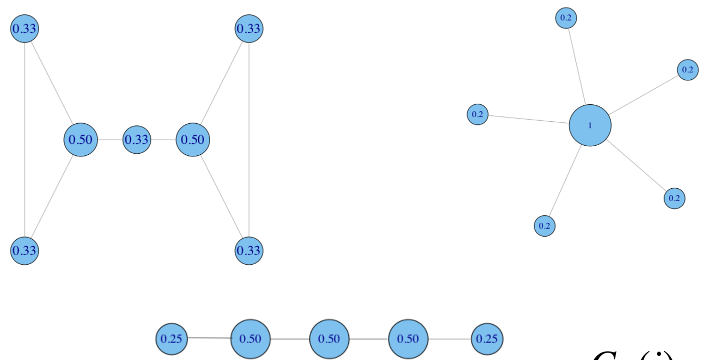
DEGREE CENTRALITY (UNDIRECTED)



$$C_D = d(i) = A_{i+} = \sum_j A_{ij}$$

87

NORMALIZED DEGREE CENTRALITY



$$C_D(i) = \frac{d(i)}{N-1}$$

88

WHEN IS DEGREE NOT SUFFICIENT?

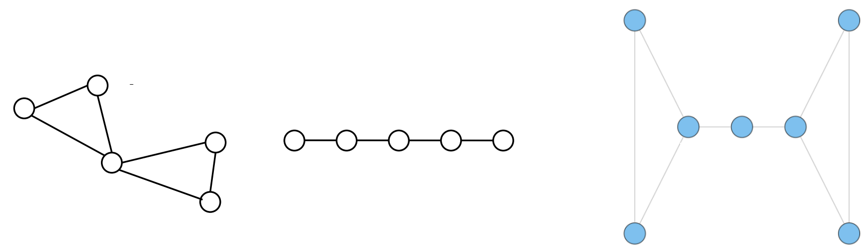
Does not capture

- Ability to broker between groups
- Likelihood that information originating anywhere in the network reaches you

89

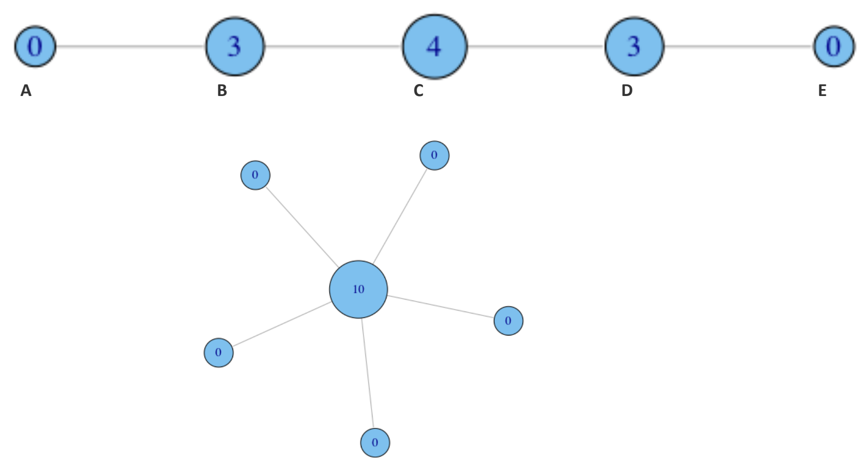
BETWEENESS

Assuming nodes communicate using the most direct (shortest) route, how many pairs of nodes have to pass information through target node?



90

BETWEENESS - EXAMPLES



91

BETWEENESS - DEFINITION

$$C_B(i) = \sum_{j,k \neq i, j<k} g_{jk}(i) / g_{jk}$$

g_{jk} = the number of shortest paths connecting *jk*
$g_{jk}(i)$ = the number of shortest paths containing *i*.

Normalization:

$$C_B^{'}(i) = C_B(i\ ) / [(n-1)(n-2)/2]$$

number of pairs of vertices
excluding the vertex itself

92

WHEN ARE C_d, AND C_b NOT SUFFICIENT?

Does not capture

Likelihood that information originating anywhere in the network reaches you

94

CLOSENESS - DEFINITION

e.g., which node is closest to the *center* of the graph

Closeness Centrality:

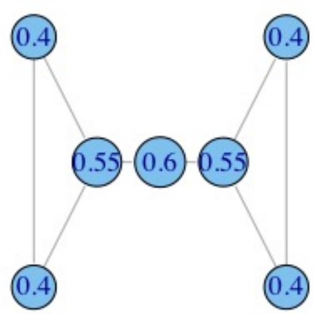
$$C_c(i) = \left[\sum_{j=1, j \neq i}^{N} d(i,j) \right]^{-1}$$

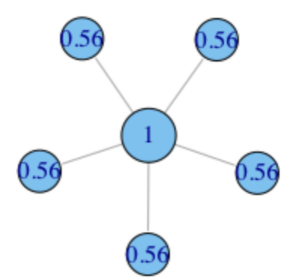
Normalized Closeness Centrality

$$C_C^{'}(i) = (C_C(i)) / (N-1) = \frac{N-1}{\sum_{j=1, j \neq i}^{N} d(i,j)}$$

95

CLOSENESS - EXAMPLES



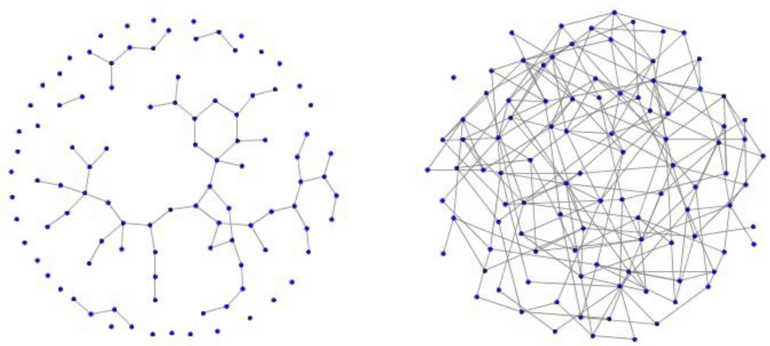


96

COMMUNITY STRUCTURE

106

HOW DENSE IS IT?



density = e/ e_{max}

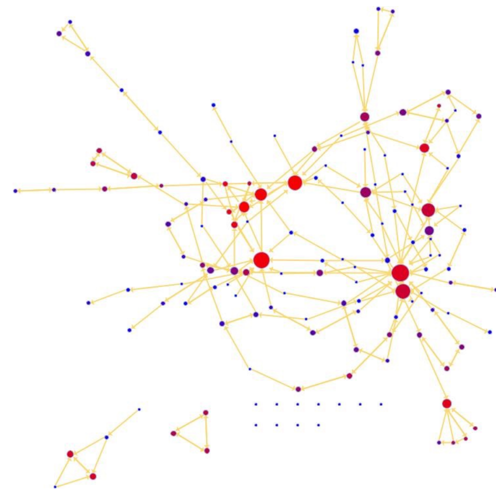
Max. possible edges:

Directed: e_{max} = n*(n-1)

Undirected: e_{max} = n*(n-1)/2

107

IS EVERYTHING CONNECTED?



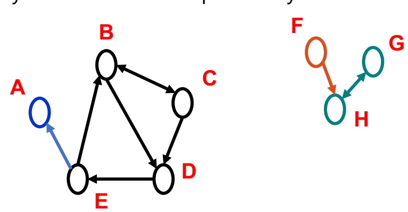
108

CONNECTED COMPONENTS - DIRECTED

Strongly connected components

Each node in component can be reached from every other node in component by following directed links

B C D E
A
G H
F



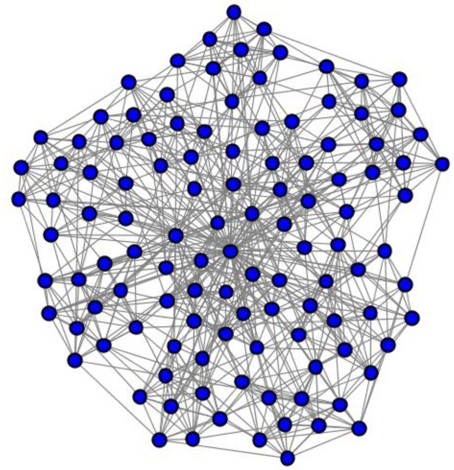
Weakly connected components

Each node can be reached from every other node by following links in either direction

A B C D E
G H F

109

COMMUNITY FINDING - CLUSTERING



112

HIERARCHICAL CLUSTERING

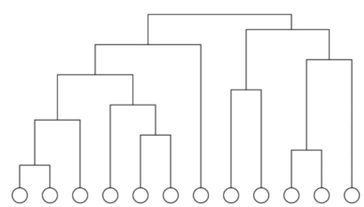
Process

Calculate affinity weights *W* for all pairs of vertices

Start: *N* disconnected vertices

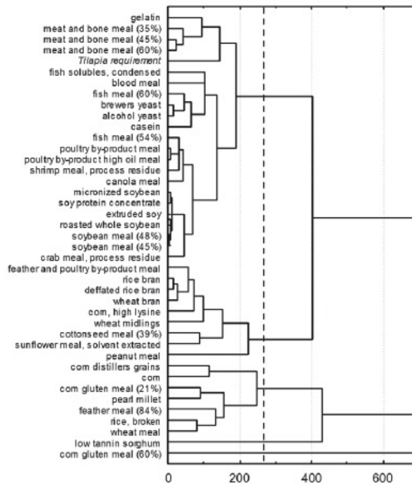
Add edges (one by one) between pairs of clusters in order of decreasing weight (use closest distance to compare clusters)

Result: nested components



113

CLUSTER DENDROGRAMS

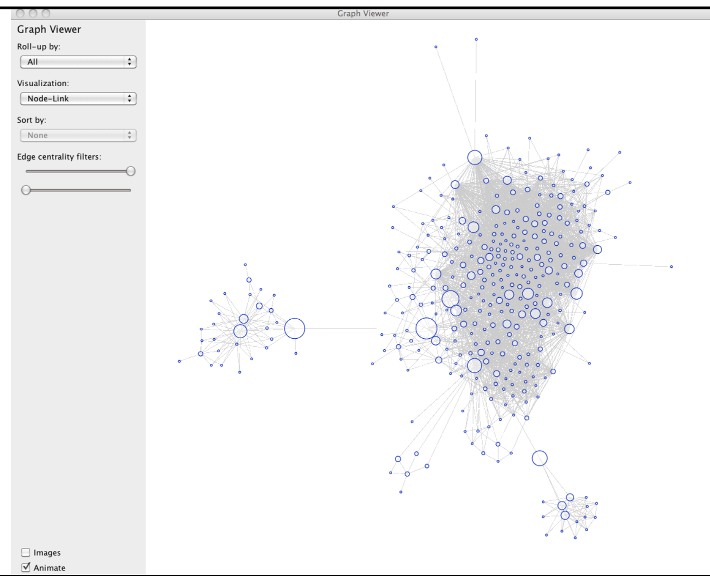


Depicts cluster trees produced by hierarchical clustering algorithms

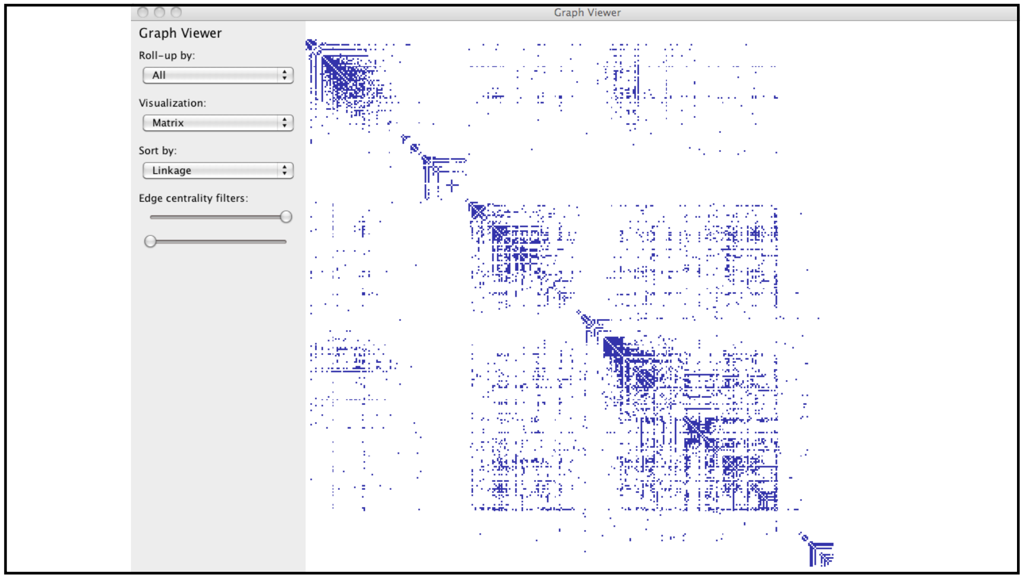
Leaf nodes arranged in line, internal node depth indicates order/value at which clusters merge

Basic recursive layout with orthogonal two-segment edges

114



115



116

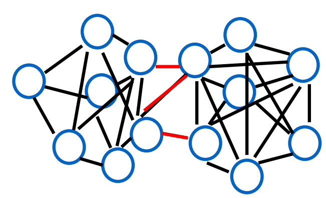
BETWEENESS CLUSTERING

Girvan and Newman 2002 iterative algorithm:

Compute C_b of all *edges*

Remove edge *i* where $C_b(i) == max(C_b)$

Recalculate betweenness



119

SIMULATING NETWORK MODELS

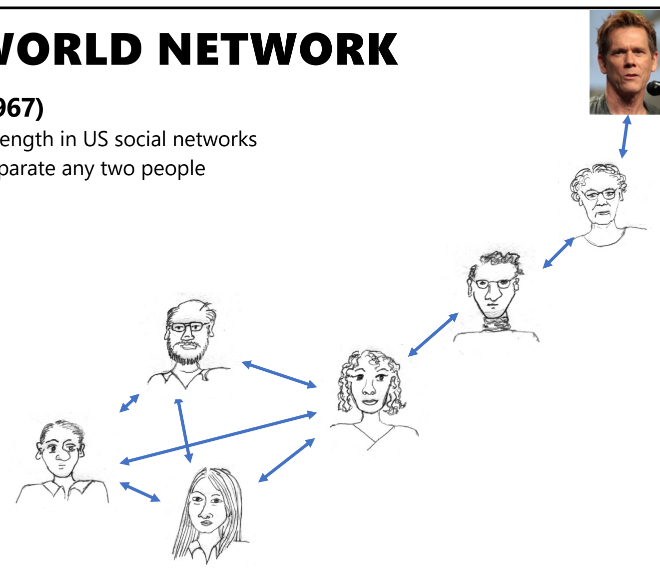
137

SMALL WORLD NETWORK

Milgram (1967)

Mean path length in US social networks

~ 6 hops separate any two people

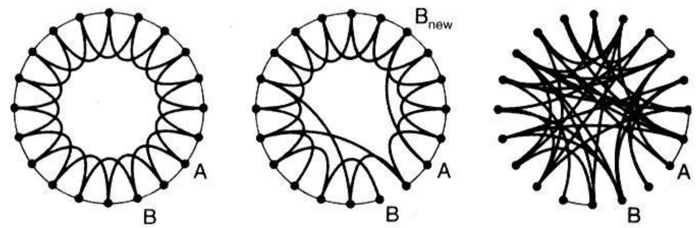


138

SMALL WORLD NETWORK

Watts and Strogatz 1998

a few random links in otherwise structured graph make network a small world



regular lattice: my friend's friend is always my friend

small world: mostly structured with a few random connections

random graph: all connections random

139

DEFINING SMALL WORLD PHENOMENA

Properties

high clustering

low mean shortest path

$$C_{\text{network}} >> C_{\text{random graph}}$$

$$l_{\text{network}} \approx \ln(N)$$

Examples

neural network of C. elegans

semantic networks of languages

actor collaboration graph

food webs

140

SUMMARY

Structural analysis
- Centrality
- Community structure

Simulation models enable further analysis

Network analysis applicable in many domains

145

146