Plan for today
- small group exercises
  - Ballotmaps
  - this week reading Q&A
    - chap Spatial papers NecklaceMaps, Myriahedral [type: algorithm]
- week 6 reading Q&A
  - chap Networks.
  - paper: Abyss-Explorer [type: design study]
  - paper: Genealogy [type: technique]
  - week 7 reading Q&A
  - paper: Polaris/Tableau [type: system]
  - paper: Vega-Lite [type: system]
  - paper: D3 [type: system]
  - paper: Abyss-Explorer [type: technique]

Next time
- reading
  - Ch 11, Manipulate View
  - Ch 12, Facet into Multiple Views
  - paper: Pattern-Driven Navigation in 2D Multiscale Visualizations with Scalable Insets [type: technique]

Q&A / Backup Slides

Information Visualization
Spatial, NecklaceMaps, Myriahedral
Ex: Ballotmaps
Tamara Munzner
Department of Computer Science
University of British Columbia

Week 8, 26 Oct 2022
https://www.cs.ubc.ca/~tmm/courses/547-22

Information Visualization
Spatial, NecklaceMaps, Myriahedral
Ex: Ballotmaps
Tamara Munzner
Department of Computer Science
University of British Columbia

Week 8, 26 Oct 2022
https://www.cs.ubc.ca/~tmm/courses/547-22

Visualization Analysis & Design
Network Data (Ch 9)
Tamara Munzner
Department of Computer Science
University of British Columbia

tamaramunzner

AcrossDistrub, Take 2
• Does it vary in different wards? Does it depend on party affiliation?

AcrossDistrub
• Does the ballot-position influence vary geographically in different boroughs?

AcrossPos
• To what extent does the position in the ballot affect the number of votes received by a candidate, overall? Is there variation across political party?

AcrossPosWithin
• To what extent does the position in the ballot affect the number of votes a candidate gets within their party?

Best of both worlds: quasi-geographic positioning
• choropleth: size issues
tabular: lose geographic position information

NameEthnicity
• To what extent does the perceived ethnicity of candidate’s name matter?

NameEthnicity
• Does this effect vary with geography?

Network data
• networks
  - model relationships between things
    - aka graphs
    - two kinds of items, both can have attributes
      - nodes
      - links
    - tree
      - special case
      - no cycles
      - one parent per node

Network tasks: topology-based and attribute-based
• topology based tasks
  - find paths
  - find (topological) neighbors
  - compare centrality/important measures
  - identify clusters / communities
• attribute based tasks (similar to table data)
  - find distributions...
• combination tasks, incorporating both
  - example: find friends-of-friends who like cats
  - topology: find all adjacent nodes of given node
  - attribute check if has-pet (node attribute) => cat

Visualization Analysis & Design
Network Data (Ch 9)
Tamara Munzner
Department of Computer Science
University of British Columbia
	@tamaramunzner

PosAcross
• To what extent does the position does the ballot affect the number of votes received by a candidate, overall? Is there variation across political party?

PosWithin
• To what extent does the position in the ballot affect the number of votes a candidate gets within their party?

AcrossDistrub, Take 2
• Does it vary in different wards? Does it depend on party affiliation?

AcrossDistrub
• Does the ballot-position influence vary geographically in different boroughs?

AcrossPos
• To what extent does the position in the ballot affect the number of votes received by a candidate, overall? Is there variation across political party?

AcrossPosWithin
• To what extent does the position in the ballot affect the number of votes a candidate gets within their party?

Best of both worlds: quasi-geographic positioning
• choropleth: size issues
tabular: lose geographic position information

NameEthnicity
• To what extent does the perceived ethnicity of candidate’s name matter?

NameEthnicity
• Does this effect vary with geography?

Network data
• networks
  - model relationships between things
    - aka graphs
    - two kinds of items, both can have attributes
      - nodes
      - links
    - tree
      - special case
      - no cycles
      - one parent per node

Network tasks: topology-based and attribute-based
• topology based tasks
  - find paths
  - find (topological) neighbors
  - compare centrality/important measures
  - identify clusters / communities
• attribute based tasks (similar to table data)
  - find distributions...
• combination tasks, incorporating both
  - example: find friends-of-friends who like cats
  - topology: find all adjacent nodes of given node
  - attribute check if has-pet (node attribute) => cat
Conflicting Criteria vs. Examples:

Symmetry

• Adjacency matrix view
  • Data: network
    - transpose into same dataset (encoding as heatmap)
  • Derived data: table from network
    - 1 quadrant
    - weighted edge between nodes
    - 2-cagon (node list x 2)
  • Visual encoding
    - cell shows presence/absence of edge
  • Scalability
    - 1K nodes, 1M edges

Node-link vs. matrix comparison

• node-link diagram strengths
  - topology understanding, path racing
  - intuitive, flexible, no training needed
• adjacency matrix strengths
  - focus on edges rather than nodes
  - layout straightforward (needing seeded)
  - predictability, scalability
  - same topology tasks tractable
• Empirical study
  - node-link best for small networks
  - matrix best for large networks

(Node Trix)

• Hybrid node/link matrix
• Capture strengths of both

Idiom: NodeTrix

- restricted node-link layouts: lay out nodes around circle or along line
- data
  - original network
  - derived node ordering attribute (global computation)
- Considerations
  - node ordering crucial to avoid excessive clutter from edge crossings
  - examples: before & after bar-centric ordering

Optimization-based layout

- formulate layout problem as optimization problem
- convert criteria into weighted function

- Force-directed placement
- Circular layouts / arc diagrams (node-link)
- Force-directed placement

- physics model
  - basis = springs pull together
  - nodes = magnets repulse apart
- Algorithm
  - place vertices in random locations
  - while not equilibrium
    - calculate forces on vertex
      - sum of...
        - pairwise repulsion of all nodes
        - attraction between connected nodes
        - move vertex by c * vertex_force
  - convergence:
    - computational expensive: O(n^3) for n nodes
  - each step is n^2, takes ~n cycles to reach equilibrium

- Node order is crucial: Reordering

- Structures visible in both

- Idiom: adjacency matrix view
  - Node: point marks
  - Links: line marks
  - Connections between nodes: intuitive & familiar

- Criteria for good node-link layouts
  - Minimize
    - edge crossings, node overlaps
    - distances between topological neighbor nodes
    - total drawing
    - edge bends
  - Maximize
    - Angular distance between different edges
    - Aspect ratio disparities
  - Emphasize Symmetry
    - Similar graph structures should look similar in layout

- Derived data: table from network
  - Data: network
  - Derive adjacency matrix from network
  - ~ restricted node-link layouts: lay out nodes around circle or along line
  - Data
    - Original network
    - Derived node ordering attribute (global computation)
  - Considerations
    - Node ordering crucial to avoid excessive clutter from edge crossings
    - Examples: before & after bar-centric ordering

- Force-directed placement
  - Physics model
    - Basis = springs pull together
    - Nodes = magnets repulse apart
  - Algorithm
    - Place vertices in random locations
    - While not equilibrium
      - Calculate forces on vertex
        - Sum of...
          - Pairwise repulsion of all nodes
          - Attraction between connected nodes
          - Move vertex by c * vertex_force
    - Convergence:
      - Computational expensive: O(n^3) for n nodes
    - Each step is n^2, takes ~n cycles to reach equilibrium

- Idiom: circular layouts / arc diagrams (node-link)
  - Restricted node-link layouts: lay out nodes around circle or along line
  - Data
    - Original network
    - Derived node ordering attribute (global computation)
  - Considerations
    - Node ordering crucial to avoid excessive clutter from edge crossings
  - Examples: before & after bar-centric ordering

- Optimization-based layout
  - Formulate layout problem as optimization problem
  - Convert criteria into weighted function
    - F(layout) = α * (crossing cost) + β * (drawing space used)
  - Use known optimization techniques to find layout at minimal cost
    - Energy-based physics models
    - Force-directed placement
    - Spring embedders

- Idiom: force-directed placement
  - Visual encoding
    - Link connection matrix, node point mark
  - Considerations
    - Spatial position: no meaning directly encoded
    - soft-to-intensive crossings
    - Proximity semantics
    - sometimes meaningful
    - Sometimes arbitrary artifact of layout algorithm
    - Tension with length
    - Long edges more visually similar than short
  - Tasks
    - Explore topology: locate paths, clusters
  - Scalability
    - Node/edge density E < 4N

- Idem: circular layouts / arc diagrams (node-link)
  - Restricted node-link layouts: lay out nodes around circle or along line
  - Data
    - Original network
    - Derived node ordering attribute (global computation)
  - Considerations
    - Node ordering crucial to avoid excessive clutter from edge crossings
  - Examples: before & after bar-centric ordering

- Optimization-based layout
  - Formulate layout problem as optimization problem
  - Convert criteria into weighted function
    - F(layout) = α * (crossing cost) + β * (drawing space used)
  - Use known optimization techniques to find layout at minimal cost
    - Energy-based physics models
    - Force-directed placement
    - Spring embedders

- Idiom: force-directed placement
  - Visual encoding
    - Link connection matrix, node point mark
  - Considerations
    - Spatial position: no meaning directly encoded
    - Soft-to-intensive crossings
    - Proximity semantics
    - Sometimes meaningful
    - Sometimes arbitrary artifact of layout algorithm
    - Tension with length
    - Long edges more visually similar than short
  - Tasks
    - Explore topology: locate paths, clusters
  - Scalability
    - Node/edge density E < 4N
**Trees**

**Idiom: treemap**
- **data**
  - tree
- **encoding**
  - area: containment marks for hierarchical structure
  - color: spatial position
  - size: “spacetime”
- **tasks**
  - query subtrees at leaf nodes
  - ex: disk space usage within filesystem
- **scalability**
  - 1M leaf nodes

**Comparison: tree drawing idioms**

**Idiom: radial-node-link tree**
- **data**
  - tree
- **encoding**
  - link: connection marks
  - point: node marks
  - radial: axis orientation
- **tasks**
  - understanding topology, following paths
- **scalability**
  - I.K. - 10K nodes (with/without labels)

**Link marks: Connection and containment**
- **marks as links (vs. nodes)**
  - common case in network drawing
- **encoding**
  - ex: all node-link diagrams
  - emphasis: topology, path tracing, aesthetic
  - semantic: tree
- **2D case: containment**
  - ex: all treemap variants
  - emphasis: area: Bases values or leaves: size coding
  - only trees

**Visualization Analysis & Design**

**Network Data (Ch 9) II**

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamaramunznerr
Hierarchical Edge Bundling

- works for any layout: treemap vs radial

Beware: Population maps trickiness!

- spurious correlations: most attributes just show where people live
- consider when to normalize by population density
- use geographic data
- should use normalized values

Visualization Analysis & Design

Spatial Data (Ch 9)

Tamara Munzner
Department of Computer Science
University of British Columbia
@tamara imaginative

Spatial data

- use given spatial position
- when?
  - dataset contains spatial attributes and they have primary importance
  - central tasks involve understanding spatial relationships
- examples
  - geographical/cartographic data
  - sensor/simulation data

Geographic Maps

Interlocking marks
- shape coded
- area coded
- position coded
- cannot encode another attribute with these channels, they’re "taken"

Thematic maps
- show spatial variability of attribute ("theme")
- combine geographic / reference map with (simple, flat) tabular data
- join together
- region: interlocking area marks (provinces, countries with outline shapes)
- site: would have point marks (sites, locations with 2D scatter clouds)
- region: categorical key attribute in table
- use to look up value attributes
- major idioms
  - choropleth
  - symbol maps
  - cartograms
  - dot density maps

Idiom: choropleth map

- use given spatial data
- when central task is understanding spatial relationships
- data
  - geographic geometry
    - table with 1 quant attribute per region
- encoding
  - position
    - use given geometry for area mark boundaries
  - color:
    - sequential segmented colormap

Idiom: sfdp (multi-level force-directed placement)

- data: compound graph
- original network
- derived cluster hierarchy stop it
- visual encoding
  - connection marks for network links
  - containment marks for hierarchy
  - point marks for nodes
- dynamic interaction
  - select individual metanodes in hierarchy to expand/contract

Beware: Population maps trickiness!

- spurious correlations: most attributes just show where people live
- consider when to normalize by population density
- vexed new data values
- used to undercount population
- but should use normalized values

Hierarchical edge bundling

- data: compound graph
- original network
- derived cluster hierarchy stop it
- visual encoding
  - connection marks for network links
  - containment marks for hierarchy
  - point marks for nodes
- dynamic interaction
  - select individual metanodes in hierarchy to expand/contract
- scalability
  - nodes, edges: IK-10K
- hard problem eventually hits

Figure 7.25: GrouseFlocks uses containment to show graph hierarchy structure
Mercator Projection

» Heavily distorts country sizes; particularly close to the poles.

Visualization Analysis & Design

Spatial Data (Ch 9) II

Choropleth maps: Recommendations
• only use when central task is understanding spatial relationships
• show only one variable at a time
• normalize when appropriate
• be careful when choosing colors & bins
• best case: regions are roughly equal sized

Choropleth map: Pros & cons
• pros
– easy to read and understand
– well established visualization (no learning curve)
– data is often collected and aggregated by geographical regions
• cons
– most effective visual variable used for geographic location
– visual salience depends on region size, not true importance wrt attribute value
– large regions appear more important than small ones
– color palette choice has a huge influence on the result

Idiom: Symbol maps
• symbol is used to represent aggregated data (mark or glyph)
– allows use of size and shape and color channels
– keep original spatial geometry in the background
– often a good alternative to choropleth maps

State population

Dot density maps: Pros & cons
• pros
– straightforward to understand
– avoids choropleth non-uniform region size problems
• cons
– challenge: normalization, just like choropleths
– show population density (correlated with attribute), not effect of interest
– perceptual disadvantage: difficult to extract quantities
– performance disadvantage: rendering many dots can be slow

Map Projections

– mathematical functions that map 3D surface geometry of the Earth to 2D maps
– all projections on plane necessarily distort surface in some way
– interactive: github.com/jwodder/mercator and jasondavies.com/maps/

Focus on Spatial

Mercator Projection

N » Heavily distorts country sizes; particularly close to the poles.

Spatial Data (Ch 9) II

Choropleth Projection

» Heavily distorts country sizes; particularly close to the poles.

Visualization Analysis & Design

Spatial Data (Ch 9) II

Choropleth maps: Recommendations
• only use when central task is understanding spatial relationships
• show only one variable at a time
• normalize when appropriate
• be careful when choosing colors & bins
• best case: regions are roughly equal sized

Choropleth map: Pros & cons
• pros
– easy to read and understand
– well established visualization (no learning curve)
– data is often collected and aggregated by geographical regions
• cons
– most effective visual variable used for geographic location
– visual salience depends on region size, not true importance wrt attribute value
– large regions appear more important than small ones
– color palette choice has a huge influence on the result

Idiom: Symbol maps
• symbol is used to represent aggregated data (mark or glyph)
– allows use of size and shape and color channels
– keep original spatial geometry in the background
– often a good alternative to choropleth maps

State population

Dot density maps: Pros & cons
• pros
– straightforward to understand
– avoids choropleth non-uniform region size problems
• cons
– challenge: normalization, just like choropleths
– show population density (correlated with attribute), not effect of interest
– perceptual disadvantage: difficult to extract quantities
– performance disadvantage: rendering many dots can be slow

Map Projections

– mathematical functions that map 3D surface geometry of the Earth to 2D maps
– all projections on plane necessarily distort surface in some way
– interactive: github.com/jwodder/mercator and jasondavies.com/maps/

Focus on Spatial
Idioms: isosurfaces, direct volume rendering

- data
  - scalar spatial field (3D volume)
  - 1 quant attribute per grid cell
- task
  - shape understanding, spatial relationships

Idioms: similarity-clustered streamlines

- data
  - vector field
  - derived data (from field)
- task
  - 3D visualization

Idioms: vector and tensor fields

- data
  - multiple attributes per cell (vector 2)
- idioms families
  - flow glyphs
  - feature flow
  - global computation to detect features

Vector fields

- empirical study tasks
  - identifying critical points, identifying their types
  - predicting where a particle starting at a specified point will end up (advection)

WebGL/OpenGL

- graphics library
  - power and flexibility, complete control for graphics
  - hardware acceleration
  - many language bindings (e.g., C++, Java [w/ JOGL])
- pros
  - big learning curve if you don't know already
  - no video support must roll your own everything
- example app: Trejsnator (OpenGL)

Processing / p5.js

- layer on top of Java/OpenGL, javascript/WebGL
- visualization esp. for artists/designers
- pros
  - great sandbox for rapid prototyping
  - huge user community, great documentation
- cons
  - poor library support

example app: MzaBee

Idiom: topographic map

- data
  - geographic geometry
  - scalar spatial field
- task
  - shape understanding, spatial relationships

Vector and tensor fields

- data
  - multiple attributes per cell (vector 2)
- idioms families
  - flow glyphs
  - geometric (linear fields)
  - texture (linear fields)

Tools

- imperative: how
  - low-level rendering: Processing, OpenGL
  - parametrized visual objects: prefuse
- also: how
  - efficiency
  - accessibility

Paper: D3 System

- study design
  - paper publication
  - website
  - model taxonomy
  - system

WebGL/OpenGL

- graphics library
  - pros
    - power and flexibility, complete control for graphics
    - hardware acceleration
    - many language bindings (e.g., C++, Java [w/ JOGL])
  - cons
    - big learning curve if you don't know already
    - no video support must roll your own everything

example app: Trejsnator (OpenGL)

3.2 Saddle Point:

Closed

Attracting trajectories

Global characteristic points

Note


Preuse

- separation: abstract data, visual form, view
- data table, network
  - visual form: layout, color, size...
- view: multiple renderers

Learning

Efficiency

Appearance

Accessibility

Example

Buffer

View

Visual Form

DATA

Processing / p5.js

- layer on top of Java/OpenGL, javascript/WebGL
- visualization esp. for artists/designers
- pros
  - great sandbox for rapid prototyping
  - huge user community, great documentation
- cons
  - poor library support

example app: MzaBee

[Fig 1: Mayer et al. MzaBee: A Multiscale Syntax Browser. Proc. InfoVis 2009]
Declarative toolkits
- imperative toolkits
  - say exactly how to do it
  - familiar programming model
  - OpenGL, prefuse...
- declarative: other possibility
  - just say what to do
  - Protovis, D3

Protovis
- declarative infovis toolkit, in Javascript
  - also later Java version
- marks with inherited properties
- runs in browser
- matches mark/channel mental model
- much more interaction, geospatial, trees...
- cons
  - not all kinds of operations supported
- example app: NapkinVis (2009 course project)

D3
- objectives
  - compatibility
  - debugging
  - performance
- related work typology
  - document transformers
  - graphics libraries
  - infovis systems
  - general note: all related work sections are a mini-taxonomy/typology!

D3 capabilities
- query-driven selection
  - selection: filtered set of elements queries from the current doc
  - also partitioning/grouping
  - operators act on selections to modify content
- data binding to scenegraph elements
- sticky: available for subsequent re-selection
- customs, filters

Protovis Validation
- wide set of old/new app examples
  - expressiveness, effectiveness, scalability
  - accessibility
- analysis with cognitive dimensions of notation
  - clearness of mapping, hidden dependencies
  - role-expressiveness visibility consistency
  - recognizability: abstraction
  - hard mental operations

InfoVis Reference Model
- conceptual model underneath design of prefuse and many other toolkits
- heavily influenced much of infovis (including nested model)
  - aka infovis pipeline, data state model

Paper: Polaris/Table System

Table Algebra :: Interactive Interface
- drag and drop actions map to formal language underneath
- partitioning using shelves
- different results for ord vs quat

Polaris
- example
  - marks: Guert chart bars
color channels: nominal /categorical
spatial position channels: country x year
and a quart

Table: Table Algebra

D3
- declarative infovis toolkit, in Javascript
- also later Java version
- marks with inherited properties
- runs in browser
- matches mark/channel mental model
- much more interaction, geospatial, trees...
- cons
  - not all kinds of operations supported
- example app: NapkinVis (2009 course project)

D3
- objectives
  - compatibility
  - debugging
  - performance
- related work typology
  - document transformers
  - graphics libraries
  - infovis systems
  - general note: all related work sections are a mini-taxonomy/typology!

D3 Features
- document transformation as atomic operation
  - scene changes in representation of scenes themselves
- immediate property evaluation semantics
- avoid confusing consequences of delayed evaluation
- validation
  - performance benchmarks
  - page loads, frame rate
  - accessibility
  - everybody has voted with their feet by now!

Polaris: Soilete, Tang, and Hanrahan
- infovis spreadsheet
  - table cells have graphical elements, not just numbers
  - wide range of channels and marks
- example
  - marks: circles
  - color channel saturation
  - size channel: area
  - partition: ease x productmanship
  - ord vs quat

Table: Table Algebra