Visualization Analysis & Design
Full-Day Tutorial

Tamara Munzner
Department of Computer Science
University of British Columbia

ACT
August 2016, Iowa City IA

http://www.cs.ubc.ca/~tmm/talks.html#vad16act

@tamaramunzner
Outline

• **Session 1 8:30-10:00am**
  Visualization Analysis Framework
  – Introduction: Definitions
  – Analysis: What, Why, How
  – Marks and Channels

• **Session 2 10:30am-12:00pm**
  Spatial Layout
  – Arrange Tables
  – Arrange Spatial Data
  – Arrange Networks and Trees

• **Session 3 1:00-2:30pm**
  Color & Interaction
  – Map Color
  – Manipulate: Change, Select, Navigate
  – Facet: Juxtapose, Partition, Superimpose

• **Session 4 3:00-4:30pm**
  Guidelines and Examples
  – Reduce: Filter, Aggregate
  – Rules of Thumb
  – Q&A

http://www.cs.ubc.ca/~tmm/talks.html#vad16act
@tamaramunzner
Defining visualization (vis)

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

Why?...
Why have a human in the loop?

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

Visualization is suitable when there is a need to augment human capabilities rather than replace people with computational decision-making methods.

• don’t need vis when fully automatic solution exists and is trusted
• many analysis problems ill-specified
  – don’t know exactly what questions to ask in advance
• possibilities
  – long-term use for end users (e.g. exploratory analysis of scientific data)
  – presentation of known results
  – stepping stone to better understanding of requirements before developing models
  – help developers of automatic solution refine/debug, determine parameters
  – help end users of automatic solutions verify, build trust
Why use an external representation?

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

- external representation: replace cognition with perception

Why depend on vision?

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

• human visual system is high-bandwidth channel to brain
  – overview possible due to background processing
    • subjective experience of seeing everything simultaneously
    • significant processing occurs in parallel and pre-attentively

• sound: lower bandwidth and different semantics
  – overview not supported
    • subjective experience of sequential stream

• touch/haptics: impoverished record/replay capacity
  – only very low-bandwidth communication thus far

• taste, smell: no viable record/replay devices
Why represent all the data?

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

• summaries lose information, details matter
  – confirm expected and find unexpected patterns
  – assess validity of statistical model

Anscombe’s Quartet

<table>
<thead>
<tr>
<th>Identical statistics</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x mean</td>
<td>9</td>
</tr>
<tr>
<td>x variance</td>
<td>10</td>
</tr>
<tr>
<td>y mean</td>
<td>7.5</td>
</tr>
<tr>
<td>y variance</td>
<td>3.75</td>
</tr>
<tr>
<td>x/y correlation</td>
<td>0.816</td>
</tr>
</tbody>
</table>

Identical statistics

x mean = 9  
x variance = 10
y mean = 7.5  
y variance = 3.75  
x/y correlation = 0.816
Why focus on tasks and effectiveness?

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks and work effectively.

- tasks serve as constraint on design (as does data)
  - idioms do not serve all tasks equally!
  - challenge: recast tasks from domain-specific vocabulary to abstract forms
- most possibilities ineffective
  - validation is necessary, but tricky
  - increases chance of finding good solutions if you understand full space of possibilities
- what counts as effective?
  - novel: enable entirely new kinds of analysis
  - faster: speed up existing workflows
What resource limitations are we faced with?

Vis designers must take into account three very different kinds of resource limitations: those of computers, of humans, and of displays.

• computational limits
  – processing time
  – system memory

• human limits
  – human attention and memory

• display limits
  – pixels are precious resource, the most constrained resource
  – information density: ratio of space used to encode info vs unused whitespace
    • tradeoff between clutter and wasting space, find sweet spot between dense and sparse
Why analyze?

• imposes structure on huge design space
  – scaffold to help you think systematically about choices
  – analyzing existing as stepping stone to designing new
  – most possibilities ineffective for particular task/data combination

SpaceTree

TreeJuxtaposer


Further reading

  – Chap 1: What’s Vis, and Why Do It?
Outline

• **Session 1 8:30-10:00am**
  Visualization Analysis Framework
  – Introduction: Definitions
  – Analysis: What, Why, How
  – Marks and Channels

• **Session 2 10:30am-12:00pm**
  Spatial Layout
  – Arrange Tables
  – Arrange Spatial Data
  – Arrange Networks and Trees

• **Session 3 1:00-2:30pm**
  Color & Interaction
  – Map Color
  – Manipulate: Change, Select, Navigate
  – Facet: Juxtapose, Partition, Superimpose

• **Session 4 3:00-4:30pm**
  Guidelines and Examples
  – Reduce: Filter, Aggregate
  – Rules of Thumb
  – Q&A

http://www.cs.ubc.ca/~tmm/talks.html#vad16act
Analysis framework: Four levels, three questions

- **domain** situation
  - who are the target users?

- **abstraction**
  - translate from specifics of domain to vocabulary of vis

- **what** is shown? **data abstraction**
  - often don’t just draw what you’re given: transform to new form

- **why** is the user looking at it? **task abstraction**

- **idiom**

- **how** is it shown?
  - visual encoding idiom: how to draw
  - interaction idiom: how to manipulate

- **algorithm**
  - efficient computation


[A Multi-Level Typology of Abstract Visualization Tasks
Why is validation difficult?

- different ways to get it wrong at each level
Why is validation difficult?

• solution: use methods from different fields at each level

   - **Domain situation**
     Observe target users using existing tools

   - **Data/task abstraction**
     - **Visual encoding/interaction idiom**
       Justify design with respect to alternatives
     - **Algorithm**
       Measure system time/memory
       Analyze computational complexity
       Analyze results qualitatively
       Measure human time with lab experiment (*lab study*)
     - Observe target users after deployment (*field study*)
   - Measure adoption

What?

Why?

How?

<table>
<thead>
<tr>
<th>Data Types</th>
<th>Attributes</th>
<th>Links</th>
<th>Positions</th>
<th>Grids</th>
</tr>
</thead>
<tbody>
<tr>
<td>Items</td>
<td>Attributes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Attributes</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data and Dataset Types</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Tables</td>
<td>Networks &amp; Trees</td>
<td>Fields</td>
<td>Geometry</td>
<td>Clusters, Sets, Lists</td>
</tr>
<tr>
<td>Items</td>
<td>Items (nodes)</td>
<td>Grids</td>
<td>Items</td>
<td>Items</td>
</tr>
<tr>
<td>Attributes</td>
<td>Links</td>
<td>Positions</td>
<td>Positions</td>
<td></td>
</tr>
<tr>
<td>Attributes</td>
<td>Attributes</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset Types</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Tables</td>
<td>Networks</td>
<td>Fields</td>
<td>Geometry</td>
<td></td>
</tr>
<tr>
<td>Multidimensional Table</td>
<td>Trees</td>
<td>(Continuous)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ordering Direction</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequential</td>
<td>Diverging</td>
<td>Cyclic</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attribute Types</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Categorical</td>
<td>Ordered</td>
<td>Ordinal</td>
<td>Quantitative</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset Availability</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Static</td>
<td>Dynamic</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Why?

How?

What?
Three major datatypes

Dataset Types

- **Tables**
  - Attributes (columns)
  - Items (rows)
  - Cell containing value
  - **Multidimensional Table**

- **Networks**
  - Link
  - Node (item)
  - **Trees**

- **Spatial**
  - Fields (Continuous)
  - Geometry (Spatial)
  - Grid of positions
  - Cell
  - Attributes (columns)
  - Value in cell
  - **Spatial Networks**

- **Visualization vs computer graphics**
  - geometry is design decision
## Dataset and data types

### Data and Dataset Types

<table>
<thead>
<tr>
<th>Tables</th>
<th>Networks &amp; Trees</th>
<th>Fields</th>
<th>Geometry</th>
<th>Clusters, Sets, Lists</th>
</tr>
</thead>
<tbody>
<tr>
<td>Items</td>
<td>Items (nodes)</td>
<td>Grids</td>
<td>Items</td>
<td>Items</td>
</tr>
<tr>
<td>Attributes</td>
<td>Links</td>
<td>Positions</td>
<td>Attributes</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Data Types

- **Items**
- **Attributes**
- **Links**
- **Positions**
- **Grids**

### Dataset Availability

- **Static**
- **Dynamic**
Attribute types

Attribute Types:
- Categorical
- Ordered
- Ordinal
- Quantitative

Ordering Direction:
- Sequential
- Diverging
- Cyclic
• \{action, target\} pairs
  – discover distribution
  – compare trends
  – locate outliers
  – browse topology
**Actions: Analyze**

- **consume**
  - discover vs present
    - classic split
    - aka explore vs explain
  - enjoy
    - newcomer
    - aka casual, social

- **produce**
  - annotate, record
  - derive
    - crucial design choice
Derive

• don’t just draw what you’re given!
  – decide what the right thing to show is
  – create it with a series of transformations from the original dataset
  – draw that

• one of the four major strategies for handling complexity

Original Data

![Graph of Original Data]

Derived Data

![Graph of Derived Data]

\[ \text{trade balance} = \text{exports} - \text{imports} \]
**Actions: Search, query**

- **what does user know?**
  - target, location

- **how much of the data matters?**
  - one, some, all

- **independent choices for each of these three levels**
  - analyze, search, query
  - mix and match

---

<table>
<thead>
<tr>
<th>Location known</th>
<th>Target known</th>
<th>Location unknown</th>
<th>Target unknown</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><em>Lookup</em></td>
<td><em>Browse</em></td>
<td></td>
</tr>
<tr>
<td><em>Locate</em></td>
<td><em>Identify</em></td>
<td></td>
<td></td>
</tr>
<tr>
<td><em>Explore</em></td>
<td><em>Compare</em></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Search**

**Query**

**Identify**

**Compare**

**Summarize**
Analysis example: Derive one attribute

- **Strahler number**
  - centrality metric for trees/networks
  - derived quantitative attribute
  - draw top 5K of 500K for good skeleton


Task 1

- **In** Tree
- **Out** Quantitative attribute on nodes

**What?**
- In Tree
- Out Quantitative attribute on nodes

**Why?**
- Derive

Task 2

- **In** Tree + **In** Quantitative attribute on nodes
- **Out** Filtered Tree
- **Out** Quantitative attribute on nodes

**What?**
- In Tree
- In Quantitative attribute on nodes
- Out Filtered Tree

**Why?**
- Summarize
- Topology

**How?**
- Reduce
- Filter

Removed unimportant parts
Why: Targets

- All Data
  - Trends
  - Outliers
  - Features

- Attributes
  - One
    - Distribution
    - Extremes
  - Many
    - Dependency
    - Correlation
    - Similarity

- Network Data
  - Topology
    - Paths

- Spatial Data
  - Shape
### How?

<table>
<thead>
<tr>
<th>Encode</th>
<th>Manipulate</th>
<th>Facet</th>
<th>Reduce</th>
</tr>
</thead>
</table>
| **Arrange**<br>→ Express<br>→ Separate | **Map**<br>→ Color<br>→ Hue<br>→ Saturation<br>→ Luminance<br>→ Size, Angle, Curvature, ...<br>→ Shape<br>→ Motion<br>  
  *Direction, Rate, Frequency, ...* | **Change**<br>**Select**<br>**Navigate**<br>**Juxtapose**<br>**Partition**<br>**Superimpose**<br>**Filter**<br>**Aggregate**<br>**Embed**<br> | **Filter**<br>**Aggregate**<br>**Embed** |
| **Order**<br>→ Align | | | |
| **Use**<br> | | | |

What? | Why? | How? |
| --- | --- | --- |
Further reading

  - Chap 2: What: Data Abstraction
  - Chap 3: Why: Task Abstraction


Outline

• **Session 1 8:30-10:00am**
  Visualization Analysis Framework
  – Introduction: Definitions
  – Analysis: What, Why, How
  – Marks and Channels

• **Session 2 10:30am-12:00pm**
  Spatial Layout
  – Arrange Tables
  – Arrange Spatial Data
  – Arrange Networks and Trees

• **Session 3 1:00-2:30pm**
  Color & Interaction
  – Map Color
  – Manipulate: Change, Select, Navigate
  – Facet: Juxtapose, Partition, Superimpose

• **Session 4 3:00-4:30pm**
  Guidelines and Examples
  – Reduce: Filter, Aggregate
  – Rules of Thumb
  – Q&A

http://www.cs.ubc.ca/~tmm/talks.html#vad16act
Visual encoding

• analyze idiom structure
Definitions: Marks and channels

- **marks**
  - geometric primitives

- **channels**
  - control appearance of marks
  - can redundantly code with multiple channels
Visual encoding

• analyze idiom structure
  – as combination of marks and channels

1: vertical position
mark: line

2: vertical position
horizontal position
mark: point

3: vertical position
horizontal position
color hue
mark: point

4: vertical position
horizontal position
color hue
size (area)
mark: point
Channels

Position on common scale
Position on unaligned scale
Length (1D size)
Tilt/angle
Area (2D size)
Depth (3D position)
Color luminance
Color saturation
Curvature
Volume (3D size)

Spatial region
Color hue
Motion
Shape
Channels: Matching Types

**Magnitude Channels: Ordered Attributes**
- Position on common scale
- Position on unaligned scale
- Length (1D size)
- Tilt/angle
- Area (2D size)
- Depth (3D position)
- Color luminance
- Color saturation
- Curvature
- Volume (3D size)

**Identity Channels: Categorical Attributes**
- Spatial region
- Color hue
- Motion
- Shape

- expressiveness principle
  - match channel and data characteristics
Channels: Rankings

**Magnitude Channels: Ordered Attributes**

- Position on common scale
- Position on unaligned scale
- Length (1D size)
- Tilt/angle
- Area (2D size)
- Depth (3D position)
- Color luminance
- Color saturation
- Curvature
- Volume (3D size)

**Identity Channels: Categorical Attributes**

- Spatial region
- Color hue
- Motion
- Shape

- **expressiveness principle**
  - match channel and data characteristics
- **effectiveness principle**
  - encode most important attributes with highest ranked channels
Channels: Expressiveness types and effectiveness rankings

**Magnitude Channels: Ordered Attributes**
- Position on common scale
- Position on unaligned scale
- Length (1D size)
- Tilt/angle
- Area (2D size)
- Depth (3D position)
- Color luminance
- Color saturation
- Curvature
- Volume (3D size)

**Identity Channels: Categorical Attributes**
- Spatial region
- Color hue
- Motion
- Shape

- **expressiveness principle**
  - match channel and data characteristics

- **effectiveness principle**
  - encode most important attributes with highest ranked channels
  - spatial position ranks high for both
Accuracy: Fundamental Theory

Steven’s Psychophysical Power Law: $S = I^N$

- Electric Shock: $N = 3.5$
- Satiuration: $N = 1.7$
- Length: $N = 1$
- Area: $N = 0.7$
- Depth: $N = 0.67$
- Brightness: $N = 0.5$
Accuracy: Vis experiments

Discriminability: How many usable steps?

- must be sufficient for number of attribute levels to show
  - linewidth: few bins

[mappa.mundi.net/maps/maps_014/telegeography.html]
Separability vs. Integrality

Position
+ Hue (Color)

- Fully separable
  - 2 groups each

Size
+ Hue (Color)

- Some interference
  - 2 groups each

Width
+ Height

- Some/significant interference
  - 3 groups total: integral area

Red
+ Green

- Major interference
  - 4 groups total: integral hue
Popout

• find the red dot
  – how long does it take?

• parallel processing on many individual channels
  – speed independent of distractor count
  – speed depends on channel and amount of difference from distractors

• serial search for (almost all) combinations
  – speed depends on number of distractors
• many channels: tilt, size, shape, proximity, shadow direction, ...
• but not all! parallel line pairs do not pop out from tilted pairs
Grouping

- containment
- connection

Marks as Links

- Containment
- Connection

Identity Channels: Categorical Attributes

- Spatial region
- Color hue
- Motion
- Shape
Relative vs. absolute judgements

• perceptual system mostly operates with relative judgements, not absolute
  – that’s why accuracy increases with common frame/scale and alignment
  – Weber’s Law: ratio of increment to background is constant
    • filled rectangles differ in length by 1:9, difficult judgement
    • white rectangles differ in length by 1:2, easy judgement

Relative luminance judgements

- perception of luminance is contextual based on contrast with surroundings

http://persci.mit.edu/gallery/checkershadow
Relative color judgements

- color constancy across broad range of illumination conditions

[Link to see for yourself](http://www.purveslab.net/seeforyourself/)
Further reading

  – Chap 5: Marks and Channels


Outline

• **Session 1 8:30-10:00am**
  Visualization Analysis Framework
  – Introduction: Definitions
  – Analysis: What, Why, How
  – Marks and Channels

• **Session 2 10:30am-12:00pm**
  Spatial Layout
  – Arrange Tables
  – Arrange Spatial Data
  – Arrange Networks and Trees

• **Session 3 1:00-2:30pm**
  Color & Interaction
  – Map Color
  – Manipulate: Change, Select, Navigate
  – Facet: Juxtapose, Partition, Superimpose

• **Session 4 3:00-4:30pm**
  Guidelines and Examples
  – Reduce: Filter, Aggregate
  – Rules of Thumb
  – Q&A

http://www.cs.ubc.ca/~tmm/talks.html#vad16act  
@tamaramunzner
**Encode**

- **Arrange**
  - Express
  - Separate
- **Order**
  - Align
- **Use**

**Map**

- from *categorical* and *ordered* attributes
- **Color**
  - Hue
  - Saturation
  - Luminance
- **Size, Angle, Curvature, ...**
- **Shape**
- **Motion**
  - Direction, Rate, Frequency, ...

**Manipulate**

- **Change**
- **Select**
- **Navigate**

**Facet**

- **Juxtapose**
- **Partition**
- **Superimpose**

**Reduce**

- **Filter**
- **Aggregate**
- **Embed**

---

**How?**
Encode tables: Arrange space

Encode

Arrange
- Express
- Order

Separate
- Align
Arrange tables

Express Values

Separate, Order, Align Regions

Separate

Order

Align

1 Key

2 Keys

3 Keys

Many Keys

List

Matrix

Volume

Recursive Subdivision

Axis Orientation

Rectilinear

Parallel

Radial

Layout Density

Dense

Space-Filling

Dense

Space-Filling

Rectilinear

Parallel

Radial

Parallel

Radial

Radial
Keys and values

• key
  – independent attribute
  – used as unique index to look up items
  – simple tables: 1 key
  – multidimensional tables: multiple keys

• value
  – dependent attribute, value of cell

• classify arrangements by key count
  – 0, 1, 2, many...

Express Values

1 Key
List

2 Keys
Matrix

3 Keys
Volume

Many Keys
Recursive Subdivision
Idiom: **scatterplot**

- *express values*
  - quantitative attributes
- *no keys, only values*
  - data
    - 2 quant attrs
  - mark: points
  - channels
    - horiz + vert position
- *tasks*
  - find trends, outliers, distribution, correlation, clusters
- *scalability*
  - hundreds of items

Some keys: Categorical regions

• **regions**: contiguous bounded areas distinct from each other
  – using space to *separate* (proximity)
  – following expressiveness principle for categorical attributes

• use ordered attribute to *order* and *align* regions

- **1 Key**
  - *List*

- **2 Keys**
  - *Matrix*

- **3 Keys**
  - *Volume*

- **Many Keys**
  - *Recursive Subdivision*
Idiom: **bar chart**

- one key, one value
  - data
    - 1 categ attrib, 1 quant attrib
  - mark: lines
  - channels
    - length to express quant value
    - spatial regions: one per mark
      - separated horizontally, aligned vertically
      - ordered by quant attrib
        » by label (alphabetical), by length attrib (data-driven)
  - task
    - compare, lookup values
  - scalability
    - dozens to hundreds of levels for key attrib
Idiom: **line chart**

- one key, one value
  - data
    - 2 quant attrs
  - mark: points
    - line connection marks between them
  - channels
    - aligned lengths to express quant value
    - separated and ordered by key attr into horizontal regions
- task
  - find trend
    - connection marks emphasize ordering of items along key axis by explicitly showing relation between one item and the next
Idiom: stacked bar chart

• one more key
  – data
    • 2 categ attrib, 1 quant attrib
  – mark: vertical stack of line marks
    • glyph: composite object, internal structure from multiple marks
  – channels
    • length and color hue
    • spatial regions: one per glyph
      – aligned: full glyph, lowest bar component
      – unaligned: other bar components
  – task
    • part-to-whole relationship
  – scalability
    • several to one dozen levels for stacked attrib

Idiom: **streamgraph**

- generalized stacked graph
  - emphasizing horizontal continuity
    - vs vertical items
- data
  - 1 categ key attrib (artist)
  - 1 ordered key attrib (time)
  - 1 quant value attrib (counts)
- derived data
  - geometry: layers, where height encodes counts
  - 1 quant attrib (layer ordering)
- scalability
  - hundreds of time keys
  - dozens to hundreds of artist keys
  - more than stacked bars, since most layers don’t extend across whole chart

Choosing bar vs line charts

- depends on type of key attrib
  - bar charts if categorical
  - line charts if ordered
- do not use line charts for categorical key attribs
  - violates expressiveness principle
    - implication of trend so strong that it overrides semantics!
      - “The more male a person is, the taller he/she is”

**Idiom: heatmap**

- two keys, one value
  - data
    - 2 categ attribs (gene, experimental condition)
    - 1 quant attrib (expression levels)
  - marks: area
    - separate and align in 2D matrix
      - indexed by 2 categorical attributes
  - channels
    - color by quant attrib
      - (ordered diverging colormap)
  - task
    - find clusters, outliers
- scalability
  - 1M items, 100s of categ levels, ~10 quant attrib levels
Axis Orientation

- Rectilinear
- Parallel
- Radial
Idioms: scatterplot matrix, parallel coordinates

• scatterplot matrix (SPLOM)
  – rectilinear axes, point mark
  – all possible pairs of axes
  – scalability
    • one dozen attribs
    • dozens to hundreds of items

• parallel coordinates
  – parallel axes, jagged line representing item
  – rectilinear axes, item as point
    • axis ordering is major challenge
  – scalability
    • dozens of attribs
    • hundreds of items

Task: Correlation

- scatterplot matrix
  - positive correlation
    - diagonal low-to-high
  - negative correlation
    - diagonal high-to-low
  - uncorrelated

- parallel coordinates
  - positive correlation
    - parallel line segments
  - negative correlation
    - all segments cross at halfway point
  - uncorrelated
    - scattered crossings


Figure 3. Parallel Coordinate Plot of Six-Dimensional Data Illustrating Correlations of $\rho = 1, .8, .2, 0, -.2, -.8, \text{ and } -1.$
Idioms: **radial bar chart, star plot**

- radial bar chart
  - radial axes meet at central ring, line mark
- star plot
  - radial axes, meet at central point, line mark
- bar chart
  - rectilinear axes, aligned vertically

- accuracy
  - length unaligned with radial
    - less accurate than aligned with rectilinear

Idioms: **pie chart, polar area chart**

- **pie chart**
  - area marks with angle channel
  - accuracy: angle/area less accurate than line length
    - arclength also less accurate than line length

- **polar area chart**
  - area marks with length channel
  - more direct analog to bar charts

- **data**
  - 1 categ key attrib, 1 quant value attrib

- **task**
  - part-to-whole judgements

Idioms: **normalized stacked bar chart**

- **task**
  - part-to-whole judgements

- **normalized stacked bar chart**
  - stacked bar chart, normalized to full vert height
  - single stacked bar equivalent to full pie
    - high information density: requires narrow rectangle

- **pie chart**
  - information density: requires large circle

[http://bl.ocks.org/mbostock/3887235](http://bl.ocks.org/mbostock/3887235)
[http://bl.ocks.org/mbostock/3886208](http://bl.ocks.org/mbostock/3886208)
[http://bl.ocks.org/mbostock/3886394](http://bl.ocks.org/mbostock/3886394)
Idiom: **glyphmaps**

- rectilinear good for linear vs nonlinear trends

- radial good for cyclic patterns

Orientation limitations

- rectilinear: scalability wrt #axes
  - 2 axes best
  - 3 problematic
    - more in afternoon
  - 4+ impossible
- parallel: unfamiliarity, training time
- radial: perceptual limits
  - angles lower precision than lengths
  - asymmetry between angle and length
  - can be exploited!

Layout Density

→ Dense


dense software overviews
Further reading

  —Chap 7: Arrange Tables


• A Brief History of Data Visualization. Friendly. 2008.
  http://www.datavis.ca/milestones
Outline

• **Session 1** 8:30-10:00am
  Visualization Analysis Framework
  – Introduction: Definitions
  – Analysis: What, Why, How
  – Marks and Channels

• **Session 2** 10:30am-12:00pm
  Spatial Layout
  – Arrange Tables
  – Arrage Spatial Data
  – Arrange Networks and Trees

• **Session 3** 1:00-2:30pm
  Color & Interaction
  – Map Color
  – Manipulate: Change, Select, Navigate
  – Facet: Juxtapose, Partition, Superimpose

• **Session 4** 3:00-4:30pm
  Guidelines and Examples
  – Reduce: Filter, Aggregate
  – Rules of Thumb
  – Q&A

http://www.cs.ubc.ca/~tmm/talks.html#vad16act
Arrange spatial data

➔ Use Given

➔ Geometry
  ➔ Geographic
  ➔ Other Derived

➔ Spatial Fields

➔ Scalar Fields (one value per cell)
  ➔ Isocontours
  ➔ Direct Volume Rendering

➔ Vector and Tensor Fields (many values per cell)
  ➔ Flow Glyphs (local)
  ➔ Geometric (sparse seeds)
  ➔ Textures (dense seeds)
  ➔ Features (globally derived)
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
  - use given geometry for area mark boundaries
  - sequential segmented colormap [more later]

http://bl.ocks.org/mbostock/4060606
Beware: Population maps trickiness!

[ https://xkcd.com/1138 ]
Idiom: **topographic map**

- data
  - geographic geometry
  - scalar spatial field
    - 1 quant attribute per grid cell
- derived data
  - isoline geometry
    - isocontours computed for specific levels of scalar values

*Land Information New Zealand Data Service*
Idioms: *isosurfaces, direct volume rendering*

• data
  – scalar spatial field
    • 1 quant attribute per grid cell

• task
  – shape understanding, spatial relationships

• isosurface
  – derived data: isocontours computed for specific levels of scalar values

• direct volume rendering
  – transfer function maps scalar values to color, opacity


Vector and tensor fields

• data
  – many attribs per cell

• idiom families
  – flow glyphs
    • purely local
  – geometric flow
    • derived data from tracing particle trajectories
    • sparse set of seed points
  – texture flow
    • derived data, dense seeds
  – feature flow
    • global computation to detect features
      – encoded with one of methods above
Vector fields

• empirical study tasks
  – finding critical points, identifying their types
  – identifying what type of critical point is at a specific location
  – predicting where a particle starting at a specified point will end up (advection)


Idiom: similarity-clustered streamlines

• data
  – 3D vector field

• derived data (from field)
  – streamlines: trajectory particle will follow

• derived data (per streamline)
  – curvature, torsion, tortuosity
  – signature: complex weighted combination
  – compute cluster hierarchy across all signatures
  – encode: color and opacity by cluster

• tasks
  – find features, query shape

• scalability
  – millions of samples, hundreds of streamlines

Further reading

  –*Chap 8: Arrange Spatial Data*


Outline

- **Session 1 8:30-10:00am**
  Visualization Analysis Framework
  - Introduction: Definitions
  - Analysis: What, Why, How
  - Marks and Channels

- **Session 2 10:30am-12:00pm**
  Spatial Layout
  - Arrange Tables
  - Arrange Spatial Data
  - Arrange Networks and Trees

- **Session 3 1:00-2:30pm**
  Color & Interaction
  - Map Color
  - Manipulate: Change, Select, Navigate
  - Facet: Juxtapose, Partition, Superimpose

- **Session 4 3:00-4:30pm**
  Guidelines and Examples
  - Reduce: Filter, Aggregate
  - Rules of Thumb
  - Q&A

http://www.cs.ubc.ca/~tmm/talks.html#vad16act  @tamaramunzner
Arrange networks and trees

- **Node–Link Diagrams**
  - Connection Marks
  - ![Node–Link Diagram]
  - NETWORKS
  - TRES

- **Adjacency Matrix**
  - Derived Table
  - ![Adjacency Matrix]
  - NETWORKS
  - TRES

- **Enclosure**
  - Containment Marks
  - ![Enclosure]
  - NETWORKS
  - TRES
Idiom: **force-directed placement**

- **visual encoding**
  - link connection marks, node point marks

- **considerations**
  - spatial position: no meaning directly encoded
    - left free to minimize crossings
  - proximity semantics?
    - sometimes meaningful
    - sometimes arbitrary, artifact of layout algorithm
    - tension with length
      - long edges more visually salient than short

- **tasks**
  - explore topology; locate paths, clusters

- **scalability**
  - node/edge density $E < 4N$

[Diagram of a network diagram with nodes and edges.]
Idiom: \textbf{sfdp} (multi-level force-directed placement)

- data
  - original: network
  - derived: cluster hierarchy atop it

- considerations
  - better algorithm for same encoding technique
    - same: fundamental use of space
    - hierarchy used for algorithm speed/quality but not shown explicitly
    - (more on algorithm vs encoding in afternoon)

- scalability
  - nodes, edges: 1K-10K
  - hairball problem eventually hits


\[\text{http://www.research.att.com/yifanhu/GALLERY/GRAPHHS/index1.html}\]
Idiom: **adjacency matrix view**

- **data:** network
  - transform into same data/encoding as heatmap
- **derived data:** table from network
  - 1 quant attrib
    - weighted edge between nodes
  - 2 categ attribs: node list $\times$ 2
- **visual encoding**
  - cell shows presence/absence of edge
- **scalability**
  - 1K nodes, 1M edges

---

![matrix view](image1)

![node-link view](image2)

**Figure 7.5:** Comparing matrix and node-link views of a five-node network.  
(a) Matrix view.  
(b) Node-link view. From [Henry et al. 07], Figure 3b and 3a.  
(Permission needed.)

Matrix views of networks can achieve very high information density, up to a limit of one thousand nodes and one million edges, just like cluster heatmaps and all other matrix views that uses small area marks.

Network matrix views can also show weighted networks, where each link has an associated quantitative value attribute, by encoding with an ordered channel such as color luminance or size.

For undirected networks where links are symmetric, only half of the matrix needs to be shown, above or below the diagonal, because a link from node A to node B necessarily implies a link from B to A. For directed networks, the full square matrix has meaning, because links can be asymmetric. Figure 7.5 shows a simple example of an undirected network, with a matrix view of the five-node dataset in Figure 7.5a and a corresponding node-link view in Figure 7.5b.

---


Connection vs. adjacency comparison

• adjacency matrix strengths
  – predictability, scalability, supports reordering
  – some topology tasks trainable

• node-link diagram strengths
  – topology understanding, path tracing
  – intuitive, no training needed

• empirical study
  – node-link best for small networks
  – matrix best for large networks
    • if tasks don’t involve topological structure!

Idiom: **radial node-link tree**

- **data**
  - tree

- **encoding**
  - link connection marks
  - point node marks
  - radial axis orientation
    - angular proximity: siblings
    - distance from center: depth in tree

- **tasks**
  - understanding topology, following paths

- **scalability**
  - 1K - 10K nodes

Idiom: **treemap**

- **data**
  - tree
  - 1 quant attrib at leaf nodes

- **encoding**
  - area containment marks for hierarchical structure
  - rectilinear orientation
  - size encodes quant attrib

- **tasks**
  - query attribute at leaf nodes

- **scalability**
  - 1M leaf nodes

Link marks: Connection and containment

- marks as links (vs. nodes)
  - common case in network drawing
  - 1D case: connection
    - ex: all node-link diagrams
    - emphasizes topology, path tracing
    - networks and trees
  - 2D case: containment
    - ex: all treemap variants
    - emphasizes attribute values at leaves (size coding)
    - only trees

Tree drawing idioms comparison

• data shown
  – link relationships
  – tree depth
  – sibling order

• design choices
  – connection vs containment link marks
  – rectilinear vs radial layout
  – spatial position channels

• considerations
  – redundant? arbitrary?
  – information density?
  • avoid wasting space

Further reading

  
  – Chap 9: Arrange Networks and Trees


Outline

• Session 1 8:30-10:00am
  Visualization Analysis Framework
  – Introduction: Definitions
  – Analysis: What, Why, How
  – Marks and Channels

• Session 2 10:30am-12:00pm
  Spatial Layout
  – Arrange Tables
  – Arrange Spatial Data
  – Arrange Networks and Trees

• Session 3 1:00-2:30pm
  Color & Interaction
  – Map Color
  – Manipulate: Change, Select, Navigate
  – Facet: Juxtapose, Partition, Superimpose

• Session 4 3:00-4:30pm
  Guidelines and Examples
  – Reduce: Filter, Aggregate
  – Rules of Thumb
  – Q&A

http://www.cs.ubc.ca/~tmm/talks.html#vad16act
@tamaramunzner
Idiom design choices: Encode

Encode

Why?

How?

What?

❖ Arrange
  ➔ Express
  ➔ Order
  ➔ Use

❖ Map
  ➔ from categorical and ordered attributes
  ➔ Color
    ➔ Hue
    ➔ Saturation
    ➔ Luminance
  ➔ Size, Angle, Curvature, ...
  ➔ Shape
  ➔ Motion
    Direction, Rate, Frequency, ...
Categorical vs ordered color

Color: Luminance, saturation, hue

• 3 channels
  – identity for categorical
    • hue
  – magnitude for ordered
    • luminance
    • saturation

• RGB: poor for encoding
• HSL: better, but beware
  – lightness ≠ luminance

Luminance
Saturation
Hue

Corners of the RGB color cube
L from HLS
All the same
Luminance values
Spectral sensitivity

![Graph showing spectral sensitivity with wavelength in nanometers on the x-axis and relative sensitivity on the y-axis. The graph includes a scale for the visible spectrum, with UV on the left and IR on the right.](image-url)
Opponent color and color deficiency

- perceptual processing before optic nerve
  - one achromatic luminance channel L
  - edge detection through luminance contrast
  - two chroma channels, R-G and Y-B axis

- “color blind” if one axis has degraded acuity
  - 8% of men are red/green color deficient
  - blue/yellow is rare
Designing for color deficiency: Check with simulator

Normal vision

Deuteranope

Protanope

Tritanope

http://rehue.net

Designing for color deficiency: Avoid encoding by hue alone

- redundantly encode
  - vary luminance
  - change shape
Color deficiency: Reduces color to 2 dimensions

Normal

Protanope

Deuteranope

Tritanope

Designing for color deficiency: Blue-Orange is safe

Bezold Effect: Outlines matter

- color constancy: simultaneous contrast effect

Color/Lightness constancy: Illumination conditions

Do they match?

Image courtesy of John McCann
Color/Lightness constancy: Illumination conditions

Image courtesy of John McCann
Colormaps

- Categorical

- Ordered
  - Sequential
  - Diverging

- Binary

- Diverging

- Categorical

Colormaps

→ Categorical
   ➔ Ordered
      ➔ Sequential
         ➔ Diverging

→ Bivariate

Colormaps

- Categorical
  - use with care!
- Ordered
  - Sequential
  - Diverging
- Bivariate

Color maps

- Categorical
  - Ordered
    - Sequential
    - Diverging
- Bivariate

• color channel interactions
  - size heavily affects salience
    • small regions need high saturation
    • large need low saturation
  - saturation & luminance: 3-4 bins max
    • also not separable from transparency

Categorical color: Discriminability constraints

- noncontiguous small regions of color: only 6-12 bins

[Cinteny: flexible analysis and visualization of synteny and genome rearrangements in multiple organisms. Sinha and Meller. BMC Bioinformatics, 8:82, 2007.]
ColorBrewer

• http://www.colorbrewer2.org
• saturation and area example: size affects salience!
Ordered color: Rainbow is poor default

• problems
  – perceptually unordered
  – perceptually nonlinear

• benefits
  – fine-grained structure visible and nameable


Ordered color: Rainbow is poor default

- **problems**
  - perceptually unordered
  - perceptually nonlinear

- **benefits**
  - fine-grained structure visible and nameable

- **alternatives**
  - large-scale structure: fewer hues


Ordered color: Rainbow is poor default

• problems
  – perceptually unordered
  – perceptually nonlinear

• benefits
  – fine-grained structure visible and nameable

• alternatives
  – large-scale structure: fewer hues
  – fine structure: multiple hues with monotonically increasing luminance [eg viridis R/python]


Ordered color: Rainbow is poor default

• problems
  – perceptually unordered
  – perceptually nonlinear

• benefits
  – fine-grained structure visible and nameable

• alternatives
  – large-scale structure: fewer hues
  – fine structure: multiple hues with monotonically increasing luminance [eg viridis R/python]
  – segmented rainbows for binned
Viridis

- colorful, perceptually uniform, colorblind-safe, monotonically increasing luminance

https://cran.r-project.org/web/packages/viridis/vignettes/intro-to-viridis.html
Map other channels

- size
  - length accurate, 2D area ok, 3D volume poor

- angle
  - nonlinear accuracy
    - horizontal, vertical, exact diagonal

- shape
  - complex combination of lower-level primitives
    - many bins

- motion
  - highly separable against static
    - binary: great for highlighting
  - use with care to avoid irritation
Angle

Sequential ordered line mark or arrow glyph

Diverging ordered arrow glyph

Cyclic ordered arrow glyph
Further reading

• Visualization Analysis and Design. Munzner. AK Peters Visualization Series, CRC Press, 2014
  –Chap 10: Map Color and Other Channels

• ColorBrewer, Brewer.
  –http://www.colorbrewer2.org

  –http://www.stonesc.com/Vis06


• https://cran.r-project.org/web/packages/viridis/vignettes/intro-to-viridis.html
Outline

• **Session 1 8:30-10:00am**
  Visualization Analysis Framework
  – Introduction: Definitions
  – Analysis: What, Why, How
  – Marks and Channels

• **Session 2 10:30am-12:00pm**
  Spatial Layout
  – Arrange Tables
  – Arrange Spatial Data
  – Arrange Networks and Trees

• **Session 3 1:00-2:30pm**
  Color & Interaction
  – Map Color
  – Manipulate: Change, Select, Navigate
  – Facet: Juxtapose, Partition, Superimpose

• **Session 4 3:00-4:30pm**
  Guidelines and Examples
  – Reduce: Filter, Aggregate
  – Rules of Thumb
  – Q&A

http://www.cs.ubc.ca/~tmm/talks.html#vad16act
**Encode**

- **Arrange**
  - Express
  - Separate
- **Order**
  - Align
- **Use**

**Map**
from categorical and ordered attributes

- **Color**
  - Hue
  - Saturation
  - Luminance
- **Size, Angle, Curvature, ...**

- **Shape**

- **Motion**
  - Direction, Rate, Frequency, ...

**How?**

<table>
<thead>
<tr>
<th>Manipulate</th>
<th>Facet</th>
<th>Reduce</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Change</strong></td>
<td><strong>Juxtapose</strong></td>
<td><strong>Filter</strong></td>
</tr>
<tr>
<td><img src="image" alt="Change" /></td>
<td><img src="image" alt="Juxtapose" /></td>
<td><img src="image" alt="Filter" /></td>
</tr>
</tbody>
</table>

- **Select**
  - ![Select](image)
- **Partition**
  - ![Partition](image)
- **Aggregate**
  - ![Aggregate](image)

- **Navigate**
  - ![Navigate](image)
- **Superimpose**
  - ![Superimpose](image)
- **Embed**
  - ![Embed](image)
How to handle complexity: 1 previous strategy + 3 more

- Derive
  - derive new data to show within view
  - change view over time
  - facet across multiple views
  - reduce items/attributes within single view

- Manipulate
  - Change
  - Select
  - Navigate

- Facet
  - Juxtapose
  - Partition
  - Superimpose

- Reduce
  - Filter
  - Aggregate
  - Embed
Manipulate

Change over Time

Select

Navigate

Item Reduction

Attribute Reduction

Zoom

Geometric or Semantic

Slice

Pan/Translate

Cut

Constrained

Project
Change over time

• change any of the other choices
  – encoding itself
  – parameters
  – arrange: rearrange, reorder
  – aggregation level, what is filtered...

  – interaction entails change
Idiom: Re-encode

System: Tableau

made using Tableau, http://tableausoftware.com
Idiom: **Reorder**

- data: tables with many attributes
- task: compare rankings

System: **LineUp**

Idiom: **Realign**

- stacked bars
  - easy to compare
    - first segment
    - total bar
- align to different segment
  - supports flexible comparison

**System: LineUp**

Idiom: **Animated transitions**

- smooth transition from one state to another
  - alternative to jump cuts
  - support for item tracking when amount of change is limited
- example: multilevel matrix views
- example: animated transitions in statistical data graphics
  - [https://vimeo.com/19278444](https://vimeo.com/19278444)
Select and highlight

- selection: basic operation for most interaction
- design choices
  - how many selection types?
    - click vs hover: heavyweight, lightweight
    - primary vs secondary: semantics (eg source/target)
- highlight: change visual encoding for selection targets
  - color
    - limitation: existing color coding hidden
  - other channels (eg motion)
  - add explicit connection marks between items
Navigate: Changing item visibility

- change viewpoint
  - changes which items are visible within view
  - camera metaphor
    - zoom
      - geometric zoom: familiar semantics
      - semantic zoom: adapt object representation based on available pixels
        » dramatic change, or more subtle one
    - pan/translate
    - rotate
      - especially in 3D
- constrained navigation
  - often with animated transitions
  - often based on selection set
Idiom: **Semantic zooming**

- visual encoding change
  - colored box
  - sparkline
  - simple line chart
  - full chart: axes and tickmarks

**System: LiveRAC**

Navigate: Reducing attributes

• continuation of camera metaphor
  – slice
    • show only items matching specific value for given attribute: slicing plane
    • axis aligned, or arbitrary alignment
  – cut
    • show only items on far slide of plane from camera
  – project
    • change mathematics of image creation
      – orthographic
      – perspective
      – many others: Mercator, cabinet, ...

Further reading

  —Chap 11: Manipulate View


Outline

• **Session 1 8:30-10:00am**
  Visualization Analysis Framework
  – Introduction: Definitions
  – Analysis: What, Why, How
  – Marks and Channels

• **Session 2 10:30am-12:00pm**
  Spatial Layout
  – Arrange Tables
  – Arrange Spatial Data
  – Arrange Networks and Trees

• **Session 3 1:00-2:30pm**
  Color & Interaction
  – Map Color
  – Manipulate: Change, Select, Navigate
  – Facet: Juxtapose, Partition, Superimpose

• **Session 4 3:00-4:30pm**
  Color & Interaction
  – Reduce: Filter, Aggregate
  – Rules of Thumb
  – Q&A

http://www.cs.ubc.ca/~tmm/talks.html#vad16act
Facet

- **Juxtapose**
  
  ![Diagram of Juxtapose]  

- **Partition**
  
  ![Diagram of Partition]  

- **Superimpose**
  
  ![Diagram of Superimpose]
Juxtapose and coordinate views

- Share Encoding: Same/Different
  - Linked Highlighting
    - ![Graphs demonstrating linked highlighting]

- Share Data: All/Subset/None
  - ![Graphs demonstrating share data]

- Share Navigation
  - ![Navigation controls]

134
Idiom: **Linked highlighting**  

- see how regions contiguous in one view are distributed within another  
  - powerful and pervasive interaction idiom  
- encoding: different  
  - *multiform*  
- data: all shared  

Idiom: **bird’s-eye maps**

- encoding: same
- data: subset shared
- navigation: shared
  - bidirectional linking

- differences
  - viewpoint
  - (size)

- overview-detail

System: **Google Maps**

Idiom: **Small multiples**

- **encoding:** same
- **data:** none shared
  - different attributes for node colors
  - (same network layout)
- **navigation:** shared

---

System: **Cerebral**

### Coordinate views: Design choice interaction

<table>
<thead>
<tr>
<th>Encoding</th>
<th>Data</th>
<th>Overview/Detail</th>
<th>Small Multiples</th>
<th>None</th>
</tr>
</thead>
<tbody>
<tr>
<td>Same</td>
<td>All</td>
<td>Redundant</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Different</td>
<td>Subset</td>
<td>Multiform</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>None</td>
<td></td>
<td>Small Multiples</td>
<td></td>
</tr>
</tbody>
</table>

- **why juxtapose views?**
  - benefits: eyes vs memory
    - lower cognitive load to move eyes between 2 views than remembering previous state with single changing view
  - costs: display area, 2 views side by side each have only half the area of one view
Why not animation?

- disparate frames and regions: comparison difficult
  - vs contiguous frames
  - vs small region
  - vs coherent motion of group

- safe special case
  - animated transitions
System: **Improvise**

- investigate power of multiple views
  - pushing limits on view count, interaction complexity
  - how many is ok?
    - open research question
- reorderable lists
  - easy lookup
  - useful when linked to other encodings

Partition into views

• how to divide data between views
  – split into regions by attributes
  – encodes association between items using spatial proximity
  – order of splits has major implications for what patterns are visible

• no strict dividing line
  – view: big/detailed
    • contiguous region in which visually encoded data is shown on the display
  – glyph: small/iconic
    • object with internal structure that arises from multiple marks
Partitioning: List alignment

- Single bar chart with grouped bars
  - Split by state into regions
    - Complex glyph within each region showing all ages
  - Compare: easy within state, hard across ages

- Small-multiple bar charts
  - Split by age into regions
    - One chart per region
  - Compare: easy within age, harder across states
Partitioning: Recursive subdivision

- split by neighborhood
- then by type
- then time
  - years as rows
  - months as columns
- color by price

- neighborhood patterns
  - where it’s expensive
  - where you pay much more for detached type

Partitioning: Recursive subdivision

- switch order of splits
  - type then neighborhood
- switch color
  - by price variation
- type patterns
  - within specific type, which neighborhoods inconsistent

System: HIVE

Partitioning: Recursive subdivision

- different encoding for second-level regions
  - choropleth maps

Partitioning: Recursive subdivision

- size regions by sale counts
  - not uniformly
- result: treemap

Superimpose layers

• *layer*: set of objects spread out over region
  – each set is visually distinguishable group
  – extent: whole view

• design choices
  – how many layers, how to distinguish?
    • encode with different, nonoverlapping channels
    • two layers achievable, three with careful design
  – small static set, or dynamic from many possible?
Static visual layering

• foreground layer: roads
  – hue, size distinguishing main from minor
  – high luminance contrast from background
• background layer: regions
  – desaturated colors for water, parks, land areas
• user can selectively focus attention
• “get it right in black and white”
  – check luminance contrast with greyscale view

Superimposing limits

• few layers, but many lines
  – up to a few dozen
  – but not hundreds

• superimpose vs juxtapose: empirical study
  – superimposed for local, multiple for global
  – tasks
    • local: maximum, global: slope, discrimination
  – same screen space for all multiples vs single superimposed

Dynamic visual layering

• interactive, from selection
  – lightweight: click
  – very lightweight: hover

• ex: 1-hop neighbors

Further reading

  —Chap 12: Facet Into Multiple Views


Outline

• **Session 1 8:30-10:00am**
  Visualization Analysis Framework
  – Introduction: Definitions
  – Analysis: What, Why, How
  – Marks and Channels

• **Session 2 10:30am-12:00pm**
  Spatial Layout
  – Arrange Tables
  – Arrange Spatial Data
  – Arrange Networks and Trees

• **Session 3 1:00-2:30pm**
  Color & Interaction
  – Map Color
  – Manipulate: Change, Select, Navigate
  – Facet: Juxtapose, Partition, Superimpose

• **Session 4 3:00-4:30pm**
  Guidelines and Examples
  – Reduce: Filter, Aggregate
  – Rules of Thumb
  – Q&A

http://www.cs.ubc.ca/~tmm/talks.html#vad16act @tamaramunzner
Reduce items and attributes

- reduce/increase: inverses
- filter
  - pro: straightforward and intuitive
    - to understand and compute
  - con: out of sight, out of mind
- aggregation
  - pro: inform about whole set
  - con: difficult to avoid losing signal
- not mutually exclusive
  - combine filter, aggregate
  - combine reduce, change, facet

<table>
<thead>
<tr>
<th>Reduce</th>
<th>Filter</th>
<th>Aggregate</th>
<th>Embed</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Reducing Items and Attributes

- Filter
  - Items
  - Attributes
- Aggregate
  - Items
  - Attributes
- Embed
Idiom: **dynamic filtering**

- item filtering
- browse through tightly coupled interaction
  – alternative to queries that might return far too many or too few

---

**System:** FilmFinder

Idiom: **DOSFA**

- attribute filtering
- encoding: star glyphs

Idiom: **histogram**

- static item aggregation
- task: find distribution
- data: table
- derived data
  - new table: keys are bins, values are counts
- bin size crucial
  - pattern can change dramatically depending on discretization
  - opportunity for interaction: control bin size on the fly
Continuous scatterplot

- static item aggregation
- data: table
- derived data: table
  - key attrs x,y for pixels
  - quant attrib: overplot density
- dense space-filling 2D matrix
- color: sequential categorical hue + ordered luminance colormap

Idiom: **scented widgets**

- augment widgets for filtering to show *information scent*
  - cues to show whether value in drilling down further vs looking elsewhere
- concise, in part of screen normally considered control panel

Idiom: **boxplot**

- static item aggregation
- task: find distribution
- data: table
- derived data
  - 5 quant attribs
    - median: central line
    - lower and upper quartile: boxes
    - lower upper fences: whiskers
  - values beyond which items are outliers
- outliers beyond fence cutoffs explicitly shown

[40 years of boxplots. Wickham and Stryjewski. 2012. had.co.nz]
Idiom: **Hierarchical parallel coordinates**

- dynamic item aggregation
- derived data: *hierarchical clustering*
- encoding:
  - cluster band with variable transparency, line at mean, width by min/max values
  - color by proximity in hierarchy

Spatial aggregation

• MAUP: Modifiable Areal Unit Problem
  – gerrymandering (manipulating voting district boundaries) is one example!

[http://www.e-education.psu.edu/geog486/l4_p7.html, Fig 4.cg.6]
Dimensionality reduction

• attribute aggregation
  – derive low-dimensional target space from high-dimensional measured space
  – use when you can’t directly measure what you care about
    • true dimensionality of dataset conjectured to be smaller than dimensionality of measurements
    • latent factors, hidden variables

Tumor Measurement Data

data: 9D measured space

→ DR

derived data: 2D target space

Malignant
Benign
Idiom: Dimensionality reduction for documents

Task 1
- In HD data
- Out 2D data
- Produce
- Derive

Task 2
- In 2D data
- Out Scatterplot Clusters & points
- Discover
- Explore
- Identify
- Encode
- Navigate
- Select

Task 3
- In Scatterplot Clusters & points
- Out Labels for clusters
- Produce
- Annotate
Further reading

  —Chap 13: Reduce Items and Attributes


Outline

• Session 1 8:30-10:00am
  Visualization Analysis Framework
  – Introduction: Definitions
  – Analysis: What, Why, How
  – Marks and Channels

• Session 2 10:30am-12:00pm
  Spatial Layout
  – Arrange Tables
  – Arrange Spatial Data
  – Arrange Networks and Trees

• Session 3 1:00-2:30pm
  Color & Interaction
  – Map Color
  – Manipulate: Change, Select, Navigate
  – Facet: Juxtapose, Partition, Superimpose

• Session 4 3:00-4:30pm
  Guidelines and Examples
  – Reduce: Filter, Aggregate
  – Rules of Thumb
  – Q&A
Rules of Thumb

• No unjustified 3D
  – Power of the plane
  – Disparity of depth
  – Occlusion hides information
  – Perspective distortion dangers
  – Tilted text isn’t legible

• No unjustified 2D

• Eyes beat memory

• Resolution over immersion

• Overview first, zoom and filter, details on demand

• Responsiveness is required

• Function first, form next
No unjustified 3D: Power of the plane

• high-ranked spatial position channels: **planar** spatial position – not depth!

> **Magnitude Channels: Ordered Attributes**

Position on common scale

Position on unaligned scale

Length (1D size)

Tilt/angle

Area (2D size)

Depth (3D position)

![Steven's Psychophysical Power Law: S= I^N](chart)

Perceived Sensation

Physical Intensity

- Electric Shock (3.5)
- Satiation (1.7)
- Length (1)
- Area (0.7)
- Depth (0.67)
- Brightness (0.5)
No unjustified 3D: Danger of depth

- we don’t really live in 3D: we see in 2.05D
  - acquire more info on image plane quickly from eye movements
  - acquire more info for depth slower, from head/body motion

We can only see the outside shell of the world
Occlusion hides information

- occlusion
- interaction complexity

Perspective distortion loses information

- perspective distortion
  - interferes with all size channel encodings
  - power of the plane is lost!

[Visualizing the Results of Multimedia Web Search Engines. Mukherjea, Hirata, and Hara. InfoVis 96]
3D vs 2D bar charts

- 3D bars never a good idea!
Tilted text isn’t legible

• text legibility
  – far worse when tilted from image plane

• further reading


No unjustified 3D example: Time-series data

- extruded curves: detailed comparisons impossible

[Cluster and Calendar based Visualization of Time Series Data. van Wijk and van Selow, Proc. InfoVis 99.]
No unjustified 3D example: Transform for new data abstraction

- derived data: cluster hierarchy
- juxtapose multiple views: calendar, superimposed 2D curves

[Cluster and Calendar based Visualization of Time Series Data. van Wijk and van Selow, Proc. InfoVis 99.]
Justified 3D: shape perception

• benefits outweigh costs when task is shape perception for 3D spatial data
  – interactive navigation supports synthesis across many viewpoints

A 3-D View of a Chart That Predicts
The Economic Future: The Yield Curve

By GREGOR ASCH and AMANDA COK. MARCH 19, 2010

Yield curve 101
The yield curve shows how much it costs the federal government to borrow money for a given amount of time, revealing the relationship between long- and short-term interest rates.

It is, inherently, a forecast for what the economy holds in the future — how much inflation there will be, for example, and how healthy growth will be over the years ahead — all embodied in the price of money today, tomorrow and many years from now.

No unjustified 3D

• 3D legitimate for true 3D spatial data
• 3D needs very careful justification for abstract data
  – enthusiasm in 1990s, but now skepticism
  – be especially careful with 3D for point clouds or networks

WEBPATH-a three dimensional Web history. Frecon and Smith. Proc. InfoVis 1999
No unjustified 2D

• consider whether network data requires 2D spatial layout
  – especially if reading text is central to task!
  – arranging as network means lower information density and harder label lookup compared to text lists

• benefits outweigh costs when topological structure/context important for task
  – be especially careful for search results, document collections, ontologies
Eyes beat memory

• principle: external cognition vs. internal memory
  – easy to compare by moving eyes between side-by-side views
  – harder to compare visible item to memory of what you saw

• implications for animation
  – great for choreographed storytelling
  – great for transitions between two states
  – poor for many states with changes everywhere
    • consider small multiples instead

 literal                      abstract
 animation                  small multiples
 show time with time         show time with space
Eyes beat memory example: Cerebral

• small multiples: one graph instance per experimental condition
  – same spatial layout
  – color differently, by condition

Why not animation?

• disparate frames and regions: comparison difficult
  – vs contiguous frames
  – vs small region
  – vs coherent motion of group

• safe special case
  – animated transitions
Change blindness

- if attention is directed elsewhere, even drastic changes not noticeable
  - door experiment
- change blindness demos
  - mask in between images
Resolution beats immersion

• immersion typically not helpful for abstract data
  – do not need sense of presence or stereoscopic 3D
• resolution much more important
  – pixels are the scarcest resource
  – desktop also better for workflow integration
• virtual reality for abstract data very difficult to justify

Overview first, zoom and filter, details on demand

• influential mantra from Shneiderman


• overview = summary
  – microcosm of full vis design problem

Query
Identify
Compare
Summarise

 Query
 Identify
 Compare
 Summarise
Responsiveness is required

• three major categories
  – 0.1 seconds: perceptual processing
  – 1 second: immediate response
  – 10 seconds: brief tasks

• importance of visual feedback
Function first, form next

• start with focus on functionality
  – straightforward to improve aesthetics later on, as refinement
  – if no expertise in-house, find good graphic designer to work with

• dangerous to start with aesthetics
  – usually impossible to add function retroactively
Further reading

  – *Chap 6: Rules of Thumb*

  – *Chap 12: We Have Time Requirements*
More Information

• this talk
  http://www.cs.ubc.ca/~tmm/talks.html#vad16act

• book page (including tutorial lecture slides)
  http://www.cs.ubc.ca/~tmm/vadbook
  – 20% promo code for book+ebook combo:
    HVN17

  – illustrations: Eamonn Maguire

• papers, videos, software, talks, courses
  http://www.cs.ubc.ca/group/infovis
  http://www.cs.ubc.ca/~tmm

Visualization Analysis & Design
Tamara Munzner
Illustrations by Eamonn Maguire

Visualization Analysis and Design.