Data Visualization (CSCI 627/490)

Design & D3

Dr. David Koop
Dataset Types

- **Tables**
  - Items (rows)
  - Attributes (columns)
  - Cell containing value
  - **Multidimensional Table**
    - Key 1
    - Key 2
    - Attributes
    - Value in cell

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

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

- **Geometry (Spatial)**
  - Position
  - **Trees**

[Munzner (ill. Maguire), 2014]
song structure and singing

Joe Carmanica recently wrote about this trend for the New York Times, arguing that it was led by Drake, who popularized the rapping-and-singing formula over the past decade.

A better benchmark for Lil Uzi Vert's word count might be those of pop artists, such as Beyonce (2,556 words), or even one of his major influences: Marilyn Manson (2,466 words).

There are also genre-bending artists. If Childish Gambino's Awaken, My Love! is less hip hop in the traditional '90s boom-bap sense, is it fair to compare it to vocabulary-dense Wu-Tang albums? Genre matters in vocabulary calculations—check out the chart below, which takes 500 random samples of 35,000 words from rock, country, and hip hop.

In short, if artists depart from hip-hop song structure, we'd expect their vocabulary to go down in the number of unique words.

That said, the results are still directionally interesting. Of the 150 artists in the dataset, let's take a look at who is on top.

<table>
<thead>
<tr>
<th>Genre</th>
<th>Unique Words</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rock</td>
<td>3,800</td>
</tr>
<tr>
<td>Country</td>
<td>2,800</td>
</tr>
<tr>
<td>Hip Hop</td>
<td>5,800</td>
</tr>
</tbody>
</table>

Sets & Lists

# of Unique Words Used in 500 Random Samples of 35,000 Lyrics from Country, Rock, Hip Hop

Raw Lyrics Data via John W. Miller

[M. Daniels, 2019]
# Categorial, Ordinal, and Quantitative

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>S</th>
<th>T</th>
<th>U</th>
</tr>
</thead>
<tbody>
<tr>
<td>Order ID</td>
<td>Order Date</td>
<td>Order Priority</td>
<td>Product Container</td>
<td>Product Base Margin</td>
<td>Ship Date</td>
</tr>
<tr>
<td>3</td>
<td>10/14/06</td>
<td>5-Low</td>
<td>Large Box</td>
<td>0.8</td>
<td>10/21/06</td>
</tr>
<tr>
<td>6</td>
<td>2/21/08</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.55</td>
<td>2/22/08</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Small Pack</td>
<td>0.79</td>
<td>7/17/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Jumbo Box</td>
<td>0.72</td>
<td>7/17/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.6</td>
<td>7/18/07</td>
</tr>
<tr>
<td>32</td>
<td>7/16/07</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.65</td>
<td>7/18/07</td>
</tr>
<tr>
<td>35</td>
<td>10/23/07</td>
<td>4-Not Specified</td>
<td>Wrap Bag</td>
<td>0.52</td>
<td>10/24/07</td>
</tr>
<tr>
<td>35</td>
<td>10/23/07</td>
<td>4-Not Specified</td>
<td>Small Box</td>
<td>0.58</td>
<td>10/25/07</td>
</tr>
<tr>
<td>36</td>
<td>11/3/07</td>
<td>1-Urgent</td>
<td>Small Box</td>
<td>0.55</td>
<td>11/3/07</td>
</tr>
<tr>
<td>65</td>
<td>3/18/07</td>
<td>1-Urgent</td>
<td>Small Pack</td>
<td>0.49</td>
<td>3/19/07</td>
</tr>
<tr>
<td>66</td>
<td>1/20/05</td>
<td>5-Low</td>
<td>Wrap Bag</td>
<td>0.56</td>
<td>1/20/05</td>
</tr>
<tr>
<td>69</td>
<td>6/4/05</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.44</td>
<td>6/6/05</td>
</tr>
<tr>
<td>69</td>
<td>6/4/05</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.6</td>
<td>6/6/05</td>
</tr>
<tr>
<td>70</td>
<td>12/18/06</td>
<td>5-Low</td>
<td>Small Box</td>
<td>0.59</td>
<td>12/23/06</td>
</tr>
<tr>
<td>70</td>
<td>12/18/06</td>
<td>5-Low</td>
<td>Small Box</td>
<td>0.82</td>
<td>12/23/06</td>
</tr>
<tr>
<td>96</td>
<td>4/17/05</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.55</td>
<td>4/19/05</td>
</tr>
<tr>
<td>97</td>
<td>1/29/06</td>
<td>3-Medium</td>
<td>Medium Box</td>
<td>0.38</td>
<td>1/30/06</td>
</tr>
<tr>
<td>129</td>
<td>11/19/08</td>
<td>5-Low</td>
<td>Large Box</td>
<td>0.37</td>
<td>11/28/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.37</td>
<td>5/9/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Medium Box</td>
<td>0.38</td>
<td>5/10/08</td>
</tr>
<tr>
<td>130</td>
<td>5/8/08</td>
<td>2-High</td>
<td>Small Box</td>
<td>0.6</td>
<td>5/11/08</td>
</tr>
<tr>
<td>132</td>
<td>6/11/06</td>
<td>3-Medium</td>
<td>Medium Box</td>
<td>0.6</td>
<td>6/12/06</td>
</tr>
<tr>
<td>132</td>
<td>6/11/06</td>
<td>3-Medium</td>
<td>Jumbo Box</td>
<td>0.69</td>
<td>6/14/06</td>
</tr>
<tr>
<td>134</td>
<td>5/1/06</td>
<td>4-Not Specified</td>
<td>Large Box</td>
<td>0.82</td>
<td>5/3/08</td>
</tr>
<tr>
<td>135</td>
<td>10/21/07</td>
<td>4-Not Specified</td>
<td>Small Pack</td>
<td>0.64</td>
<td>10/23/07</td>
</tr>
<tr>
<td>135</td>
<td>10/21/07</td>
<td>4-Not Specified</td>
<td>Small Box</td>
<td>0.55</td>
<td>9/14/07</td>
</tr>
<tr>
<td>193</td>
<td>8/6/06</td>
<td>1-Urgent</td>
<td>Medium Box</td>
<td>0.57</td>
<td>8/10/06</td>
</tr>
<tr>
<td>194</td>
<td>4/5/08</td>
<td>3-Medium</td>
<td>Wrap Bag</td>
<td>0.42</td>
<td>4/7/08</td>
</tr>
</tbody>
</table>

**quantitative**

**ordinal**

**categorical**
3.1. Mathematical description and types of spirals

A spiral is easy to describe and understand in polar coordinates, i.e. in the form \( r = f(\phi) \). The distinctive feature of a spiral is that \( f \) is a monotone function. In this work we assume a spiral is described by

Several simple functions \( f \) lead to well-known types of spirals:

- **Archimedes’ spiral** has the form \( r = a \phi \). It has the special property that a ray emanating from the origin crosses two consecutive arcs of the spiral in a constant distance.

- The **Hyperbolic spiral** has the form \( r = \frac{a}{\phi} \). It is the inverse of Archimedes’ spiral with respect to the origin.

- More generally, spirals of the form \( r = a \phi^k \) are called Archimedean spirals.

- The **logarithmic spiral** has the form \( r = a e^{b \phi} \). It has the special property that all arcs cut a ray emanating from the origin under the same angle.

For the visualization of time-dependent data Archimedes’ spiral seems to be the most appropriate. In most applications data from different periods are equally important. This should be reflected visually in that the distance to other periods is always the same.

3.2. Mapping data to the spiral

In general, markers, bars, and line elements can be used to visualize time-series data similar to standard point, bar, and line graphs on Spiral Graphs. For instance, quantitative, discrete data can be presented as bars on the spiral or by marks with a corresponding distance to the spiral. However, since the \( x \) and \( y \) coordinate are needed to achieve the general form of the spiral their use is limited for the display of data values. One might consider to map data values to small absolute changes in the radius, i.e.

Yet, we have found this way of visualizing to be ineffective. We conclude that the general shape of the spiral should be untouched and other attributes should be used, such as:

- colour,
- texture, including line styles and patterns.

[Figure 1: Two visualizations of sunshine intensity using about the same screen real estate and the same color coding scheme. In the spiral visualization it is much easier to compare days, to spot cloudy time periods, or to see events like sunrise and sunset.]
Tasks

What? Why? How?

<table>
<thead>
<tr>
<th>Actions</th>
<th>Targets</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analyze</td>
<td>All Data</td>
</tr>
<tr>
<td>→ Consume</td>
<td>→ Trends</td>
</tr>
<tr>
<td>→ Discover</td>
<td></td>
</tr>
<tr>
<td>→ Present</td>
<td></td>
</tr>
<tr>
<td>→ Enjoy</td>
<td></td>
</tr>
<tr>
<td>Produce</td>
<td>Attributes</td>
</tr>
<tr>
<td>→ Annotate</td>
<td>→ One</td>
</tr>
<tr>
<td></td>
<td>→ Distribution</td>
</tr>
<tr>
<td></td>
<td>→ Dependency</td>
</tr>
<tr>
<td>Search</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Network Data</td>
</tr>
<tr>
<td></td>
<td>→ Topology</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Query</td>
<td>Spatial Data</td>
</tr>
<tr>
<td></td>
<td>→ Shape</td>
</tr>
</tbody>
</table>

[Muñzner (ill. Maguire), 2014]
Actions: Analyze

→ Consume
  → Discover
  → Present
  → Enjoy

→ Produce
  → Annotate
  → Record
  → Derive

[Munzner (ill. Maguire), 2014]
Visualization for Consumption

Discover

Present

Enjoy

[M. Stefaner, M. Wattenberg]
Memorability

Low Quality Description

High Quality Description

FORGETTABLE

[MEMORABLE]

Figure 4: Utility of visualizations in mobile datasets.

[Some text from the diagram]

[M. Borkin et al., InfoVis 2015]
Memorability of Visualizations

• S. Few: "Visualizations don’t need to be designed for memorability – they need to be designed for comprehension. For most visualizations, the comprehension that they provide need only last until the decision that it informs is made. Usually, that is only a matter of seconds."

• B. Jones (paraphrased): People make decisions using visualizations but this isn't instantaneous like robots or algorithms; they often chew on a decision for a while.

• R. Kosara: there are cases where people benefit from remembering a visualization (e.g. health-related visualization)

• Are there tradeoffs between the characteristics?
Assignment 2

- Link
- Three parts: table, horizontal bar chart, vertical bar chart
  - data processing
  - highlighting (CSCI 627)
- Vertical chart can be tricky
- Start early!
- Questions?
Visualization for Production

- Generate new material
- Annotate
- Record
- Derive (Transform)
Annotation: Circle Annotations

[S. Lu, 2017]
Record: Provenance of MTA Data Exploration
Derived Data

Original Data

Derived Data

trade balance = \textit{exports} - \textit{imports}

[Munzner (ill. Maguire), 2014]
Visualization for Production

- Generate new material
- Annotate:
  - Add more to a visualization
  - Usually associated with text, but can be graphical
- Record:
  - Persist visualizations for historical record
  - Provenance (graphical histories): how did I get here?
- Derive (Transform):
  - Create new data
  - Create derived attributes (e.g. mathematical operations, aggregation)
Actions: Search

• What does a user know?
  - Lookup: check bearings
  - Locate: find on a map
  - Browse: what’s nearby
  - Explore: where to go
  - Patterns

<table>
<thead>
<tr>
<th></th>
<th>Target known</th>
<th>Target unknown</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location known</td>
<td><img src="images/lookup_icon" alt="Lookup" /></td>
<td><img src="images/browse_icon" alt="Browse" /></td>
</tr>
<tr>
<td>Location unknown</td>
<td><img src="images/locate_icon" alt="Locate" /></td>
<td><img src="images/explore_icon" alt="Explore" /></td>
</tr>
</tbody>
</table>

[Munzner (ill. Maguire), 2014]
• Number of targets: One, Some (Often 2), or All
• Identify: characteristics or references
• Compare: similarities and differences
• Summarize: overview of everything

[Muñzner (ill. Maguire), 2014]
Targets

- **ALL DATA**
  - Trends
  - Outliers
  - Features

- **ATTRIBUTES**
  - One
    - Distribution
    - Extremes
  - Many
    - Dependency
    - Correlation
    - Similarity

- **NETWORK DATA**
  - Topology
    - Paths

- **SPATIAL DATA**
  - Shape

[Muňner (ill. Maguire), 2014]
Roadmap

- **What?** → Data
  - Types
  - Semantics
- **Why?** → Tasks
  - Actions
  - Targets
- **How** → Vis Idioms/Techniques
  - Data Representation
  - Visual Encoding
  - Interaction Encoding
Analysis Example: Different “Idioms”

[SpaceTree, Grosjean et al.]

[TreeJuxtaposer, Munzner et al.]
"Idiom" Comparison

**SpaceTree**

**TreeJuxtaposer**

**What?**
- Tree

**Why?**
- Actions
  - Present
  - Locate
  - Identify

- Targets
  - Path between two nodes

**How?**
- **SpaceTree**
  - Encode
  - Navigate
  - Select
  - Filter
  - Aggregate

- **TreeJuxtaposer**
  - Encode
  - Navigate
  - Select
  - Arrange

[Munzner (ill. Maguire), 2014]
Analysis Example: Derivation

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


Task 1

- In Tree
- Derive
- Out Quantitative attribute on nodes

Task 2

- In Tree
- Summarize
- Topology
- Reduce
- Filter
d3.js

Data-Driven Documents
Data-Driven Documents (D3)

• Open-Source JavaScript Library
• http://d3js.org/
• Original Authors: Mike Bostock, Vadim Ogievestky, and Jeff Heer
• Focus on Web standards, customization, and usability
• Grew from work on Protovis: more standard, more interactive
• By nature, a low-level library; you have control over all elements and styles
• A top project on GitHub (over 93,000 stars as of Sept. 2020)
• Lots of impressive examples
  - Bostock was a New York Times Graphics Editor
D3 Key Features

- Supports data as a core piece of Web elements
  - Loading data
  - Dealing with changing data (joins, enter/update/exit)
  - **Correspondence** between data and DOM elements
- Selections (similar to CSS) that allow greater manipulation
- Method Chaining
- Integrated layout algorithms, axes calculations, etc.
- Focus on interaction support
  - Straightforward support for transitions
  - Event handling support for user-initiated changes
• Ogievetsky has put together a nice set of interactive examples that show off the major features of D3
• http://dakoop.github.io/IntroD3/
  - (Updated from original for D3 v6)
• https://beta.observablehq.com/@dakoop/d3-intro
• Other references:
  - Murray’s book on Interactive Data Visualization for the Web
  - The D3 website: d3js.org
  - Ros's Slides on v4: https://iros.github.io/d3-v4-whats-new/
D3 Data Joins

- Two groups: data and visual elements
- Three parts of the join between them: enter, update, and exit
- enter: `s.enter()`, update: `s`, exit: `s.exit()`
Merge vs. Join

• Merge creates a new selection that includes the items from both selections
  - If you want to update all elements (including those just added via enter), use merge!
  - Useful when enter+update have similar transitions

• Join allows you to modify different parts of the selection in a single statement
  - Also will create the final selection
  - Does enter+append and exit+remove automatically
  - Pass functions to modify the enter, update, and exit parts of the selection
Transitions

- Nested transitions (those that "hang off" of a parent transition) follow immediately after the parent transition