Big Data Visual Analytics (CS 661)

Instructor: Soumya Dutta

Department of Computer Science and Engineering

Indian Institute of Technology Kanpur (IITK)

email: soumyad@cse.iitk.ac.in

Acknowledgements

- Some of the following slides are adapted from the excellent course materials and tutorials made available by:
  - Prof. Michelle Borkin (Northeastern University)

Assignment 1 - Due: 16/02/25 11:59pm

- Part 1: Simplified isocontour algorithm for 2D data without handling marching square cases or ambiguity cases explicitly
  - Traverse the cell vertices in counterclockwise order
  - Not allowed to use VTK's contour filter, write your own code following the method we discussed in class
  - You do not have to implement the entire marching squares algorithm
    - If we see you implemented marching squares, even if your code works, we will deduct points
  - You do not have to handle marching squares cases
  - This is a simplified version of the algorithm
- Part 2: VTK Volume Rendering, Transfer Function, and Shading
  - Consult VTK's manual, examples for help
- **Read the instructions very carefully!!**
  - **If you do not follow instructions, points will be deducted**

Assignment 1 - Submission Process

- Submission through HelloIITK
- Only one group member needs to submit from each group
- Submit Python scripts in a single Zipped file
- README.txt file is mandatory with detailed instructions of how to run your code and pass parameters and anything else you want the TA to know for running your code
  - If we cannot run your code, you will not get points
- There is a 10% penalty each day after the submission deadline for up to 20% (2 late days).
- After that, you get **zero**. **No exception.**
- No deadline extension!

Academic Integrity for Assignments

- We will perform plagiarism check of your codes
- If we find plagiarism, heavy punishment will be followed
  - You will get 0 and could be reported to institute

How to Say Nothing with Scientific Visualization

- Never include a color legend
- Avoid annotation
- Never mention error characteristics
- When in doubt, smooth
- Avoid providing performance data
- Never learn anything about the data or the discipline
- Never compare with others
- Never cite references of data
- Claim generalizability but show result on a single data
- Use view angle to hide shortcomings
- ‘This is easily extended to 3D’

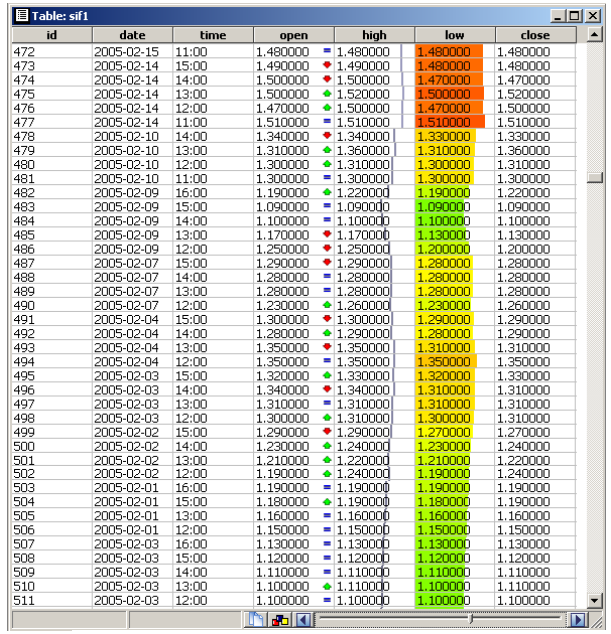
Fourteen Ways to Say Nothing with Scientific Visualization, Eric Raible, NASA, in IEEE Computer.

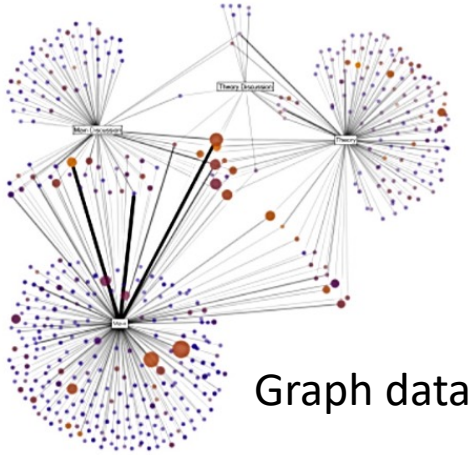
Information Visualization

Information Visualization (InfoVis)

- The use of computer-supported, interactive visual representations of data to amplify cognition
  - Data is not necessarily defined on a spatial domain
  - Data is not always numerical
  - Data is inherently discrete
- The study of transforming data, information, and knowledge into interactive visual representations

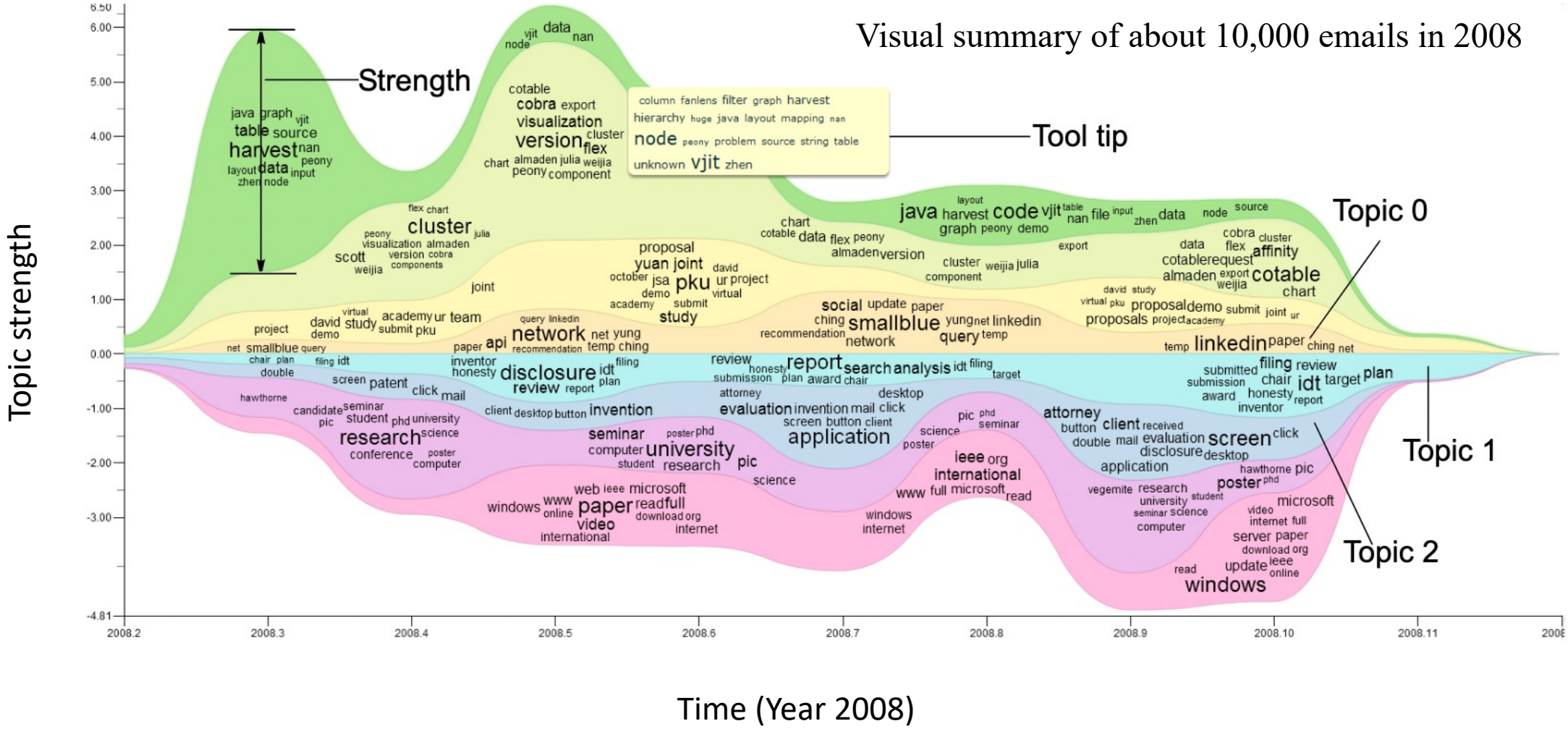
Table data



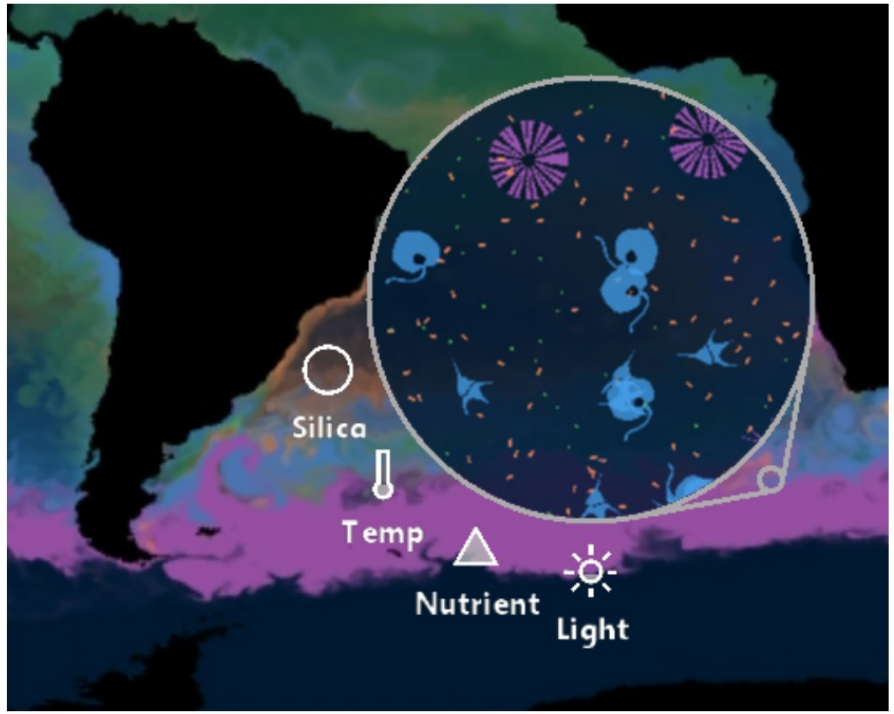


Graph data

Information Visualization for Business Data



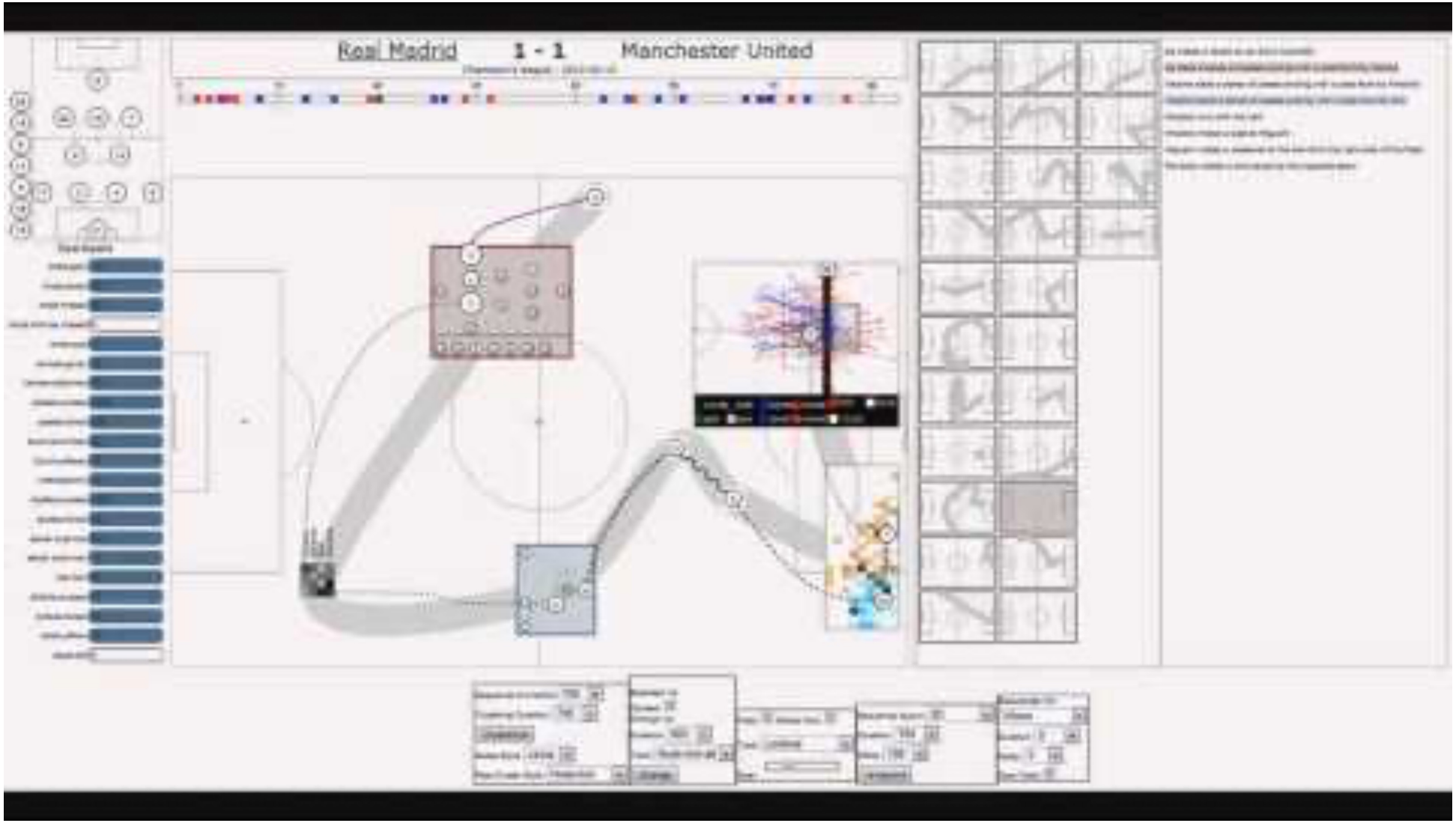
Information Visualization for Science Data



Circle viewer with indicators of environmental variables at the selected location. Silica: inorganic SiO_2 concentration; Temp: temperature; Nutrient: inorganic NO_3 concentration; Light: photosynthetically available radiation.

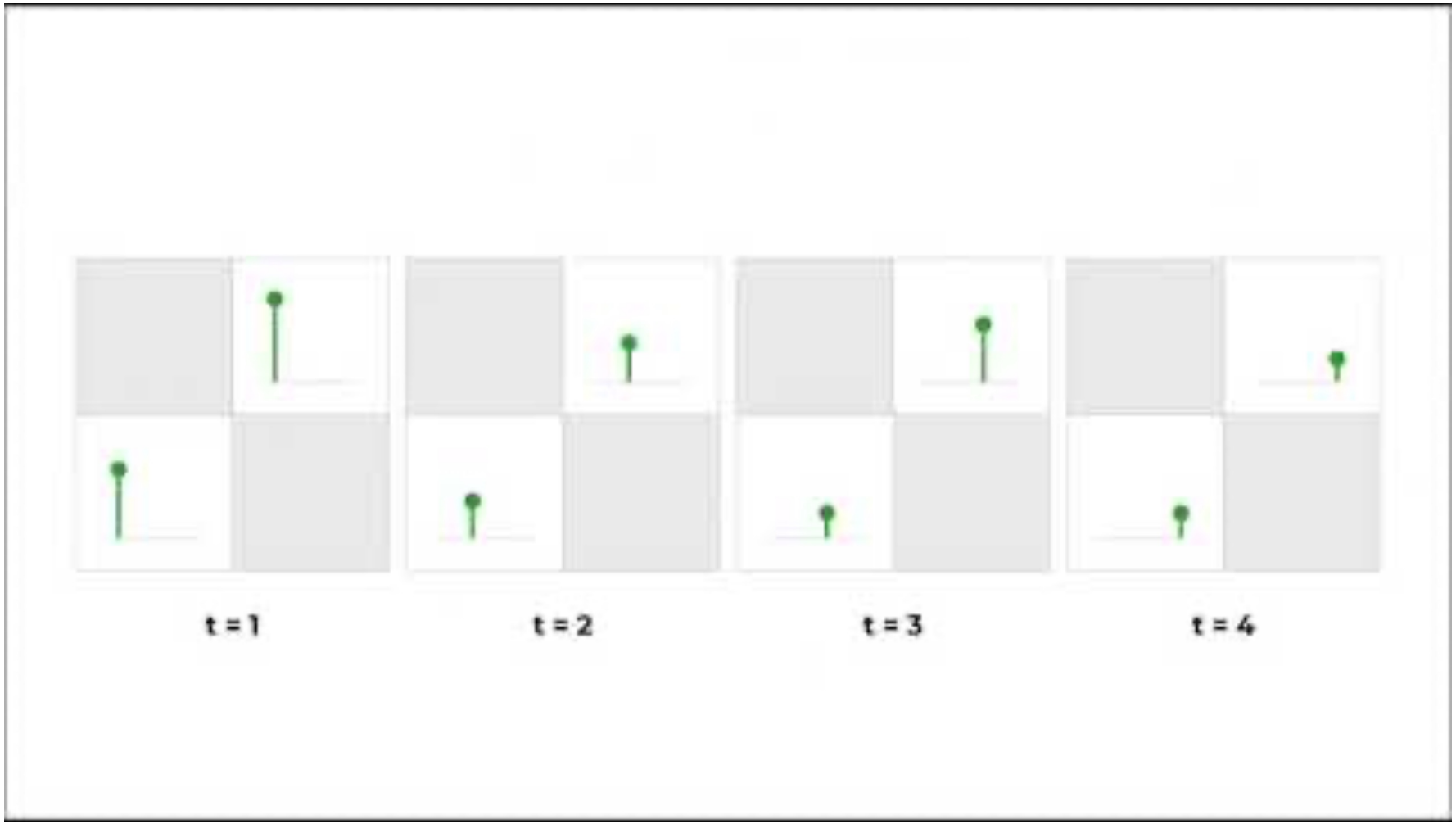
Exploratory Data Visualization Tool for Museum Visitors

Information Visualization for Soccer Data



Information Visualization for ML Classifiers

- A detailed evaluation of classifiers for model selection and debugging
- An interactive, comparative, model agnostic visualization system



Information Visualization for ML Model Explainability

A Brief Taxonomy of InfoVis Techniques

- InfoVis Techniques
  - Empirical Methods
  - Interaction
  - Frameworks
  - Applications

Empirical Methods

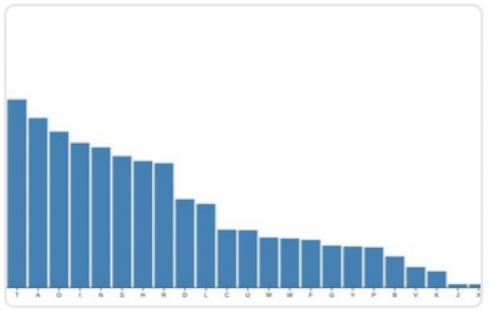
- Empirical methods are categorized as
  - Model and Evaluation
- Model
  - Visual representation model
  - Data driven model
- Evaluation
  - User studies are the most used in InfoVis and offer a scientifically sound method to measure visualization performance
    - Statistical methods

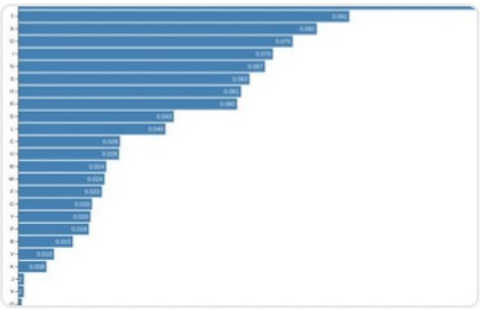
Interaction

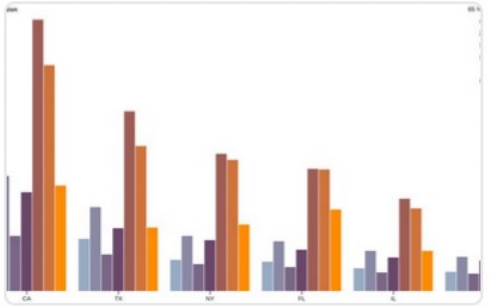
- Interaction is a fundamental aspect of InfoVis techniques
- Two Interaction categories
  - WIMP (windows, icons, mouse, pointer)
  - Post-WIMP
    - Touch interfaces
- Another operation-based categorization of interactions
  - select, explore, reconfigure, encode, abstract/elaborate, filter, and connect

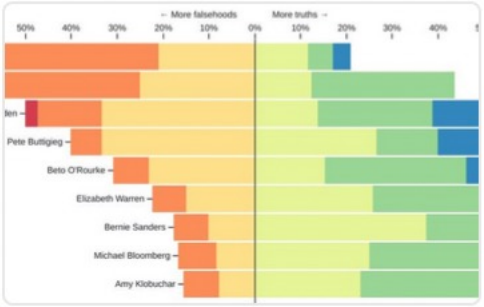
Frameworks/Systems

- Researchers have proposed a variety of visualization systems such as *Improvise*, the *InfoVis Toolkit*, and *Prefuse* to support the creation and customization of visualization applications.
- More recently, a new web-based library called Data-Driven Documents (D3) has become a very popular toolkit to construct interactive visualizations on the web
  - https://d3js.org/

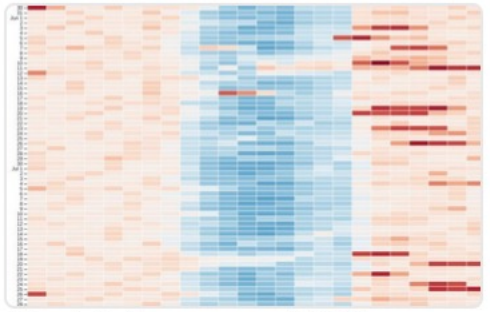

Bar chart

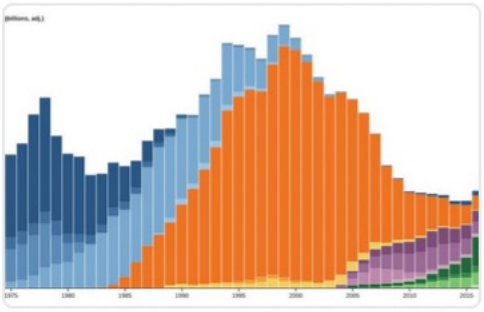

Horizontal bar chart


Grouped bar chart



Diverging stacked bar chart


Electricity usage, 2019


Revenue by music format, 1...

https://observablehq.com/@d3/galleryhttps://observablehq.com/@d3/gallery

Applications

- Four different types of data and applications
  - Graph data visualization
  - Text data visualization
  - Map data visualization
  - Multivariate data visualization

Exploratory Data Analysis

"The greatest value of a picture is when it forces us to notice what we never expected to see."

- John Tukey

InfoVis: Big Data Aspects

- Common objectives for big data visualization
  - Decision initiation or modification
  - Enhancing understanding
- Considerations for creating big data visualization systems
  - Source data
  - Information transfer to the audience
  - Design choices/scalability
- Enhance visualization by Graphical overlays
  - Highlights
  - Encodings
  - Summary statistics
  - Annotations

InfoVis: Issues and Risks

- Imprecision and Inaccuracy
  - Display information at a lower level of precision and accuracy than numerical or tabular formats
- Optical Significance
  - Viewer can interpret a difference or pattern as meaningful based on their perception, sometimes without corresponding quantitative evidence to support this interpretation
- Visualization Oversaturation
  - A dramatic increase in deficient and flawed visualizations

Libraries for Data analysis and Visualization

Libraries for Data Visualization: Matplotlib

- The most basic and Python's standard data visualization library
- A comprehensive library for creating static, animated, and interactive visualizations in Python.
- https://matplotlib.org/
- Examples: https://matplotlib.org/stable/gallery/index.html

Libraries for Data Visualization: Seaborn

- Built on top of Matplotlib but with better aesthetics and interactivity
- It provides a high-level interface for drawing attractive and informative statistical graphics.
- https://seaborn.pydata.org/
- Examples: https://seaborn.pydata.org/examples

Libraries for Data Visualization: Bokeh

- Bokeh is a Python library for creating interactive visualizations for modern web browsers.
  - Build beautiful graphics, ranging from simple plots to complex dashboards
- Create JavaScript-powered visualizations without writing any JavaScript code
- https://docs.bokeh.org/en/latest/

If you are new to Bokeh

Follow these guides to get started:

- First steps: simple tutorials that walk you through installing Bokeh and creating your first visualizations.
- User guide: explanations of all key functionalities of Bokeh and how to use them. Includes several standalone examples.

If you have some basic knowledge of Bokeh

Learn more by exploring examples:

- Gallery: a collection of examples with source code.
- Interactive tutorial notebooks: a collection of interactive notebooks to experiment with all elements of Bokeh.
- User guide: explanations of all key functionalities of Bokeh and how to use them, including examples.

If you need more advanced information

Get to know every aspect of Bokeh:

- Reference guide: detailed information about all of Bokeh's components.
- Contributor guide: information on the various ways you can contribute to the Bokeh project.

Libraries for Data Visualization: Plotly Dash

- Dash is an Open-Source Python library for creating reactive, Web-based applications
  - Built on top of Plotly.js and React.js
  - User interface library for creating analytical web applications
- https://dash.plotly.com/
- https://dash.gallery/Portal/

- **Dash is 'React' for Python**
  - React: A JavaScript library for building user interfaces

Libraries for Data Visualization: D3

- D3 - Data-Driven Documents
  - D3.js is a JavaScript library for manipulating documents based on data.
  - D3 helps you bring data to life using HTML, SVG, and CSS.
  - D3's emphasis on web standards gives you the full capabilities of modern browsers
  - Combines powerful visualization components and a data-driven approach to DOM manipulation
- https://d3js.org/