What is Data Visualization?

* Ask: What is data visualization?
* No one knows, yet is everywhere
  + Not exactly true; but wanted to start with the questions or problems around defining data visualization
  + When we think of DV - Where do we start? Histories of data visualization don’t always agree
  + We can think about the early timekeeping devices (water clocks, sand clocks)
    - Or consider the 1400s: mechanical clock in Europe; closer to the visualization of abstractions that we use to measure our world
  + We can think about charts and plots; by 1786; late 18th century; the line, area and bar charts as we know it were invented
  + Google ngram viewer (corpus of books): 1974 is when Data Visualization emerges in the corpus
  + When we speak of “Data Visualization” we are really referring to some kind of graphical display of information; not limited to the digital medium; but whose boundaries are often defined by them
  + The field is open to what current practitioners are doing and contributing; very much an open field to be pushed and defined
* Let’s move forward to what you get when you search for the definition of “Data Visualization” online
  + Broad, but I like it
  + Not very helpful?
  + How about learning to know it when you see it
* Play a Game! Guess the Data Visualization example
  + Plots
    - Statistical representations of information
    - Used to manage and make decisions in industry, governance
    - Curated messages/storytelling
  + Google Search
    - database
    - “shell” interface
  + Software
    - .db files
    - Application that reads
    - What happens when you “optimize storage”
* Last but not least, we can understand what Data Visualization by the fields that are kind of the reason why it’s alive or needed today
  + Data science: big data; working with large amounts of data requires us to find what’s most important; data visualization helps with navigating it, exploring it, making what’s most important possible to see and make decisions from
  + Information design: OG pillar of data visualization
  + Interaction Design: rise of responsive design, for different screens and the different affordances that those environments offer us – we can now make things that people can touch, can watch and carry with them everywhere; it changes our behavior vs. graphs in a print newspaper about employment rates
  + Visual Design: challenges the way that we can design data abstractions as digital artifacts
  + Computer/Information science: critical part of the data visualization stack to be able to analyze data, find patterns, have meaningful content to visualize
  + Art – data artists

Process and Principles

Process of critique is essential to the data visualization design and build process. Walk through the Design process then the development stack. I’ll show you two data visualization examples and we can critique using this framework, as well as some principles we can use to “judge” a data visualization.

Research & Research

* Human-centered design (HCR) research process
  + Problem/content: Frame the problem/hypothesis or message
    - Research
    - Explore precedents
  + Ideation: brainstorm solutions
  + Iterative prototyping
    - Low fidelity prototypes: hand-drawn sketches, wireframes
    - High fidelity prototypes: digital renders, motion mock-ups
  + User testing & feedback loops: feedback for iterative design
  + Product deliverable
  + Maintenance, ongoing testing, support and updates

Development

* Stack
  + Back-end
    - Data architecture (i.e. conceptual, diagrams, sketches)
    - Data sourcing (i.e. python, node.js, APIs)
    - Data cleaning (i.e. node.js, python, r)
    - Data storage (i.e. AWS: DynamoDB, PostgreSQL)
    - Data analysis (i.e. Python, R)
    - Data transformations (ETL software, Python, R)
    - Data maintenance (the above for each task)
  + Design
    - Research (i.e. market research, writing, reports)
    - Visual Design (i.e. Illustrator, Sketch)
    - User experience (UX) Design (i.e. Illustrator, Sketch)
    - Motion and Interaction Design (i.e. AfterEffects, Principle)
  + Build; Release
    - Front end-development (HTML/CSS, JavaScript, mapbox/leaflet, node.js, JS libraries such as d3.js, chart.js, three.js)
    - UX Engineering (HTML/CSS, JavaScript, JS frameworks such as Vue, React, Angular)
    - Production-level development (HTML/CSS, JavaScript, shadow DOM, own libraries, etc.)
  + Process can take 3 months for one release

Affordances

* Spatial: How do I learn to navigate digital abstractions as artifacts?
* Encyclopedic: How is information organized, accessed, shaped, made transparent?
* Procedural: What does it do?
* Participatory: What can I do?

My additions:

* Data democratization

An affordance of data visualizations is that data can be more accessible, can be cleaner, more transparent, and updated more frequently or in real-time

* + Fidelity/Uncertainty
  + Integrity
  + Performance
* Storytelling

An affordance of data visualizations is that data can be crafted in beautiful and interesting ways that are relevant to the viewer and targeted audiences

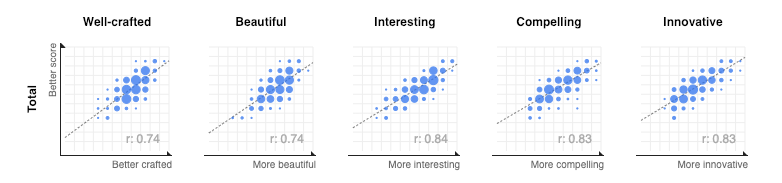
* + How simple is the story?
  + How is the viewer introduced to key concepts?
  + How is the viewer led or guided through the story?
  + How is the interactor engaging in dialogue or reciprocal exchange?
* Design

An affordance of data visualizations is that there are many visual abstractions available for expression.

* + Why were aesthetic choices made?
  + Did design support or hinder data message?
  + Did design support or hinder story?
* Motion & Interactivity

An affordance of data visualizations is that interactivity can engage interactors in dialogue or reciprocal exchange

* + Is motion or interactivity meaningful to the data message, storytelling, and/or support primary design choices?
  + Does motion or interactivity create a dynamic experience for the viewer?
  + Do motion or interactive artifacts create opportunities for dialogue or reciprocal exchange with the data?



Source: https://medium.com/@jcukier/judging-the-iib-awards-865851ff0ce8

JFI Visualizations (Precedents and Current State for each)

1. UBI
   1. Interactive clustering using a force layout
   2. Low fidelity prototype 🡪 Raw data 🡪 Python 🡪 d3.js; physics available in the force layout
2. ISA App
   1. Web app that uses components to filter data based on states
   2. Low fidelity prototype 🡪 Raw data 🡪 Python 🡪 javascript
3. Algorithmic Fairness
   1. Gamified interactive
   2. Storyboard 🡪 JavaScript