Data Visualisation - 90529

00 - Intro

Enrico Puppo
Department of Computer Science, Bioengineering, Robotics and Systems Engineering
University of Genova

21 September 2020

https://2020.aulaweb.unige.it/course/view.php?id=4293

Credits:
material in these slides is partially taken from
- T. Munzner, University of British Columbia
- A. Lex, University of Utah

other credits in the slides

Course organisation

How to follow the course

- Online lectures
  - via Teams channel code **f4dtruk**
  - presence will be registered (impact on final exam)
  - during lecture you must:
    - keep your camera **on**
    - keep your microphone **off** unless you need to ask questions or you're asked to answer questions
- In class lectures
  - room and starting date to be decided
  - dedicated to exercises, analysis of examples, design critiques, etc. Interactive!
  - you can also follow online (same rules as above)
- All lectures will be recorded and made available on my Stream channel

Required knowledge

- How to set up a web server on your PC (Apache - localhost)
- HTML, CSS, Javascript - just the basics
- general programming skills (but programming style here quite different from standard)

- if you don't have the required knowledge:
  - follow first 4 weeks of course SAW (undergraduate CS)
  - alternatively, for HTML and CSS, take the online course on eduOpen (in Italian):
    - https://learn.eduopen.org/eduopenv2/course_details.php?courseid=198
    - less thorough than SAW but fast and it should be enough to start
  - Read Chapter 3 of IDV book
- in a couple of weeks we will start using HTML, CSS and Javascript

Readings

- textbooks:
  - Tamara Munzner. Visualization Analysis and Design. AK Peters Visualization Series. CRC Press, 2014.
  - Scott Murray. Interactive Data Visualization for the Web. O'Reilly, 2013-2017 (1st-2nd edition)
- usually one chapter per class session

Exam

- Participation
  - Presence during lectures (registered)
- Homework [20%]
  - About four assigned along the course - small work, strict deadlines
- Project [50%]
  - Assigned during the course - big work, completed by the end of the course
- Oral [30%]
  - After submitting the project, by appointment for groups of students
  - Depth of oral exam proportional to % of participation:
    - >=80% : just discussion of the project
    - >=50%, <80% : discussion of the project and theory related to the project
    - < 50% : discussion of the project and all the course syllabus

Projects

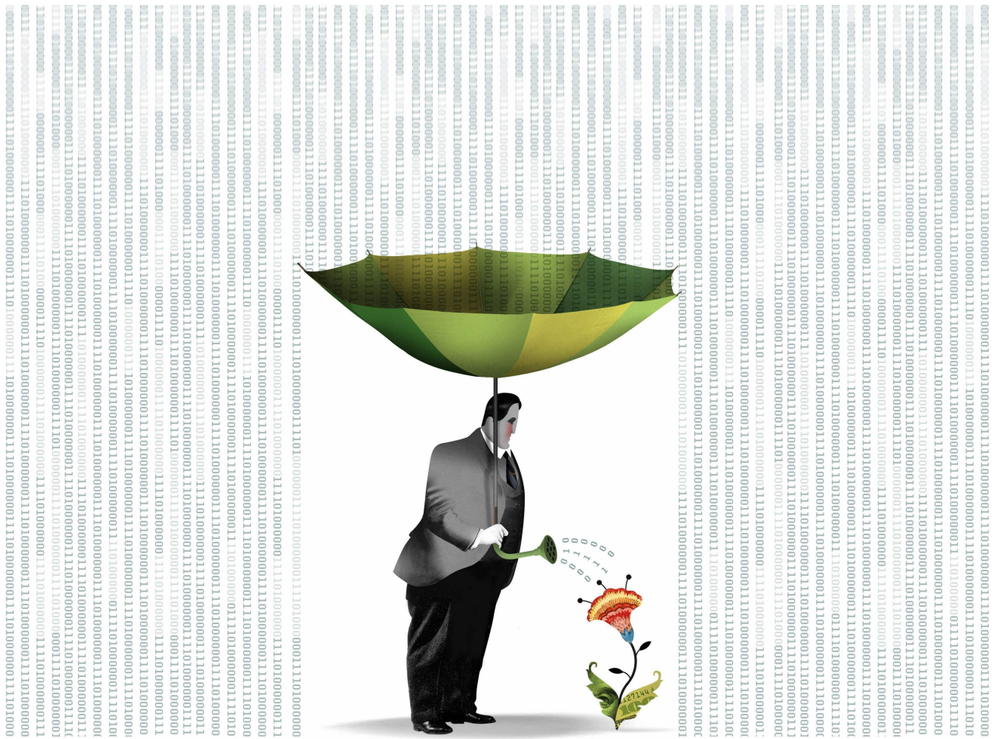
- solo, or group of 2-3
  - groups highly encouraged; amount of work commensurate with group size
- programming: develop your own Vis application
- stages
  - pitches (via Aulaweb)
  - proposal sketches (written & slides, presented in class)
  - final presentations (during oral exam)
- resources
  - software, data
  - project ideas

Finding me

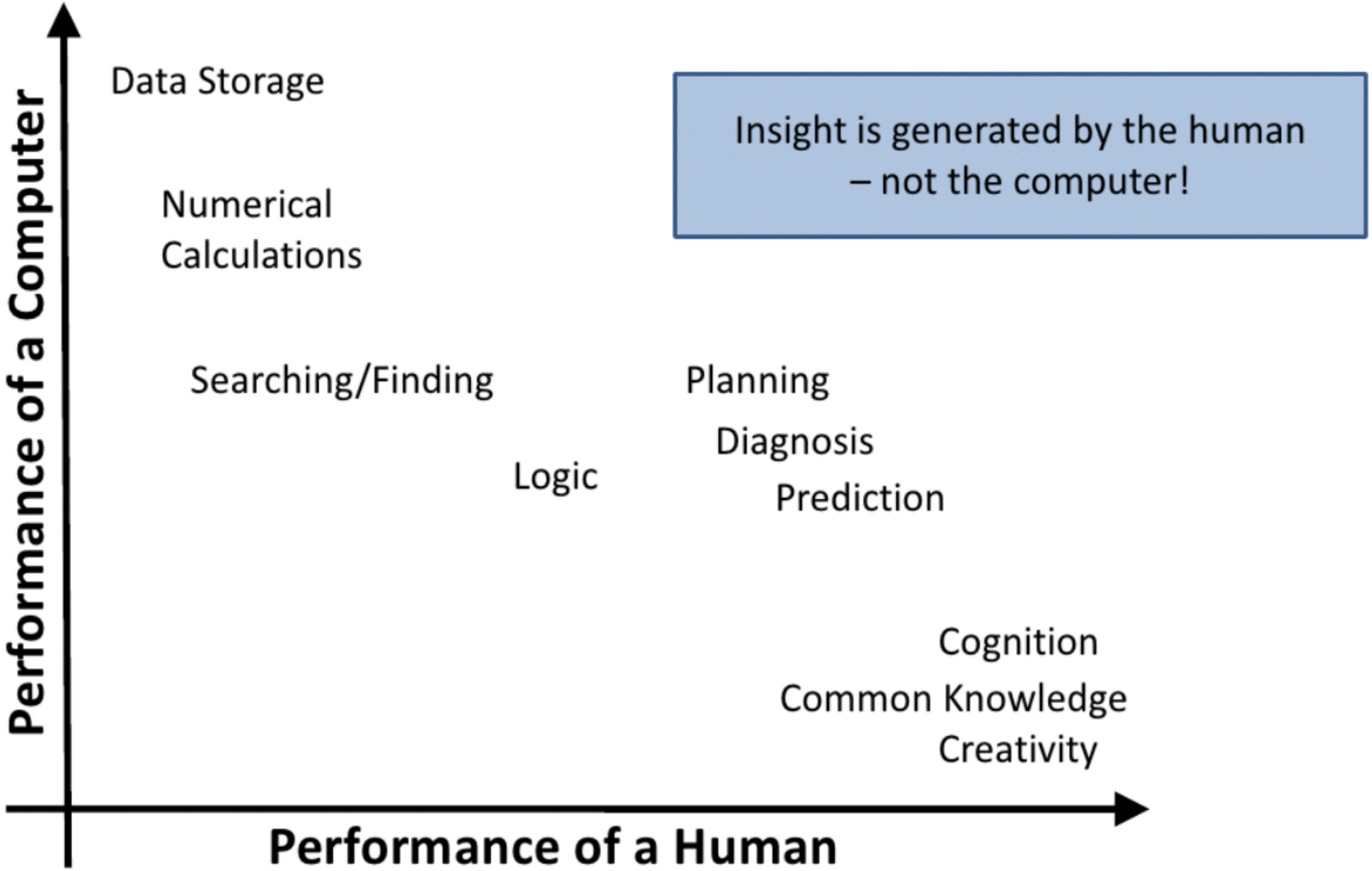
- email is the best way to reach me: enrico.puppo@unige.it
- in presence, by appointment
  - Room 306 (Valletta Puggia, floor 3, South wing)
- course page is font of all information
  - frequent updates
  - https://2020.aulaweb.unige.it/course/view.php?id=4293

Data science

- Data are many
- Often too many to easily make a sense out of them
- Sometimes computers can process them to understand their meaning
- Some other times it's better to let humans see into them



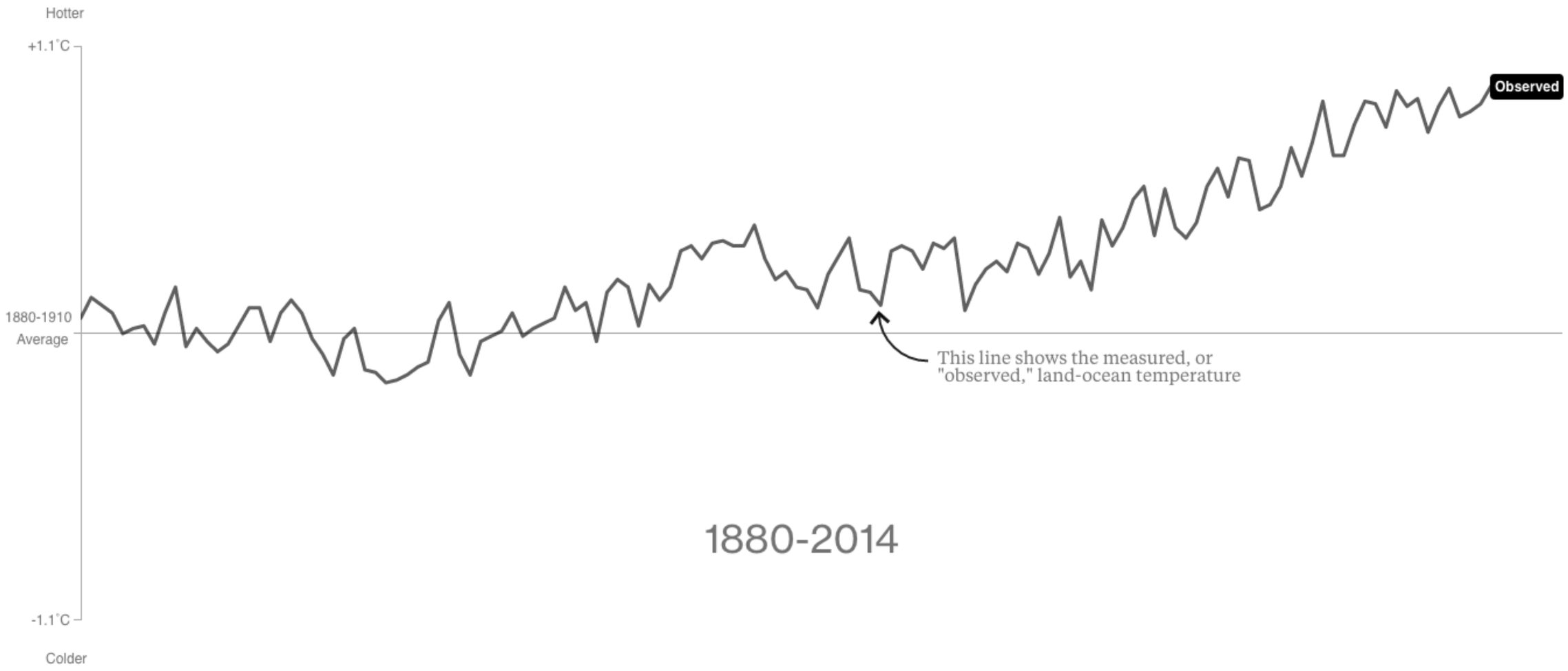
The Ability Matrix



Insight is generated by the human – not the computer!

An example

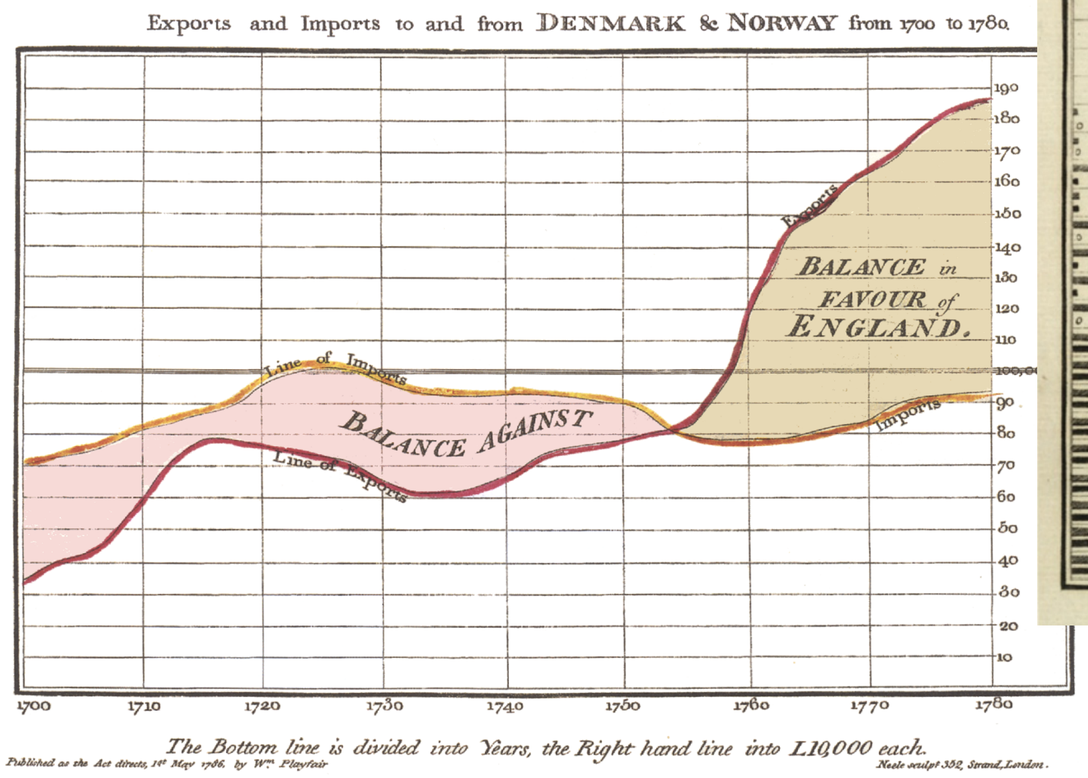
- What's really warming the world?



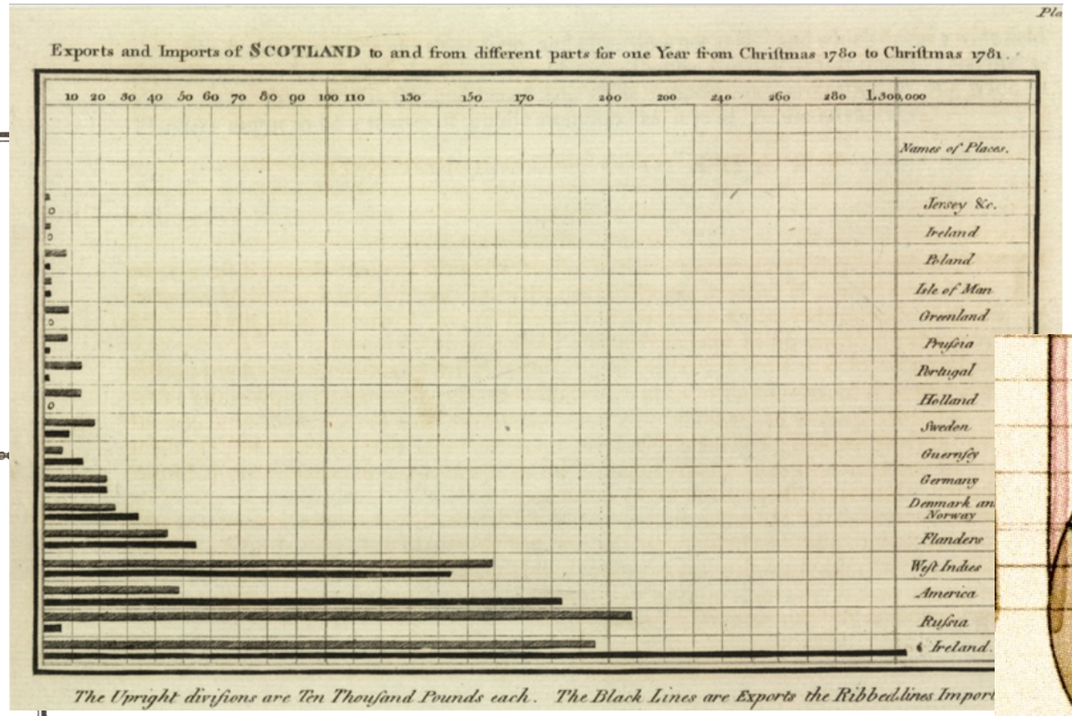
Topics Preview

Defining data visualization (vis)

- **Visual representation of facts, events, elements and their attributes that are encoded by data**
- **Not necessarily connected with computer science: data visualization exists since centuries**

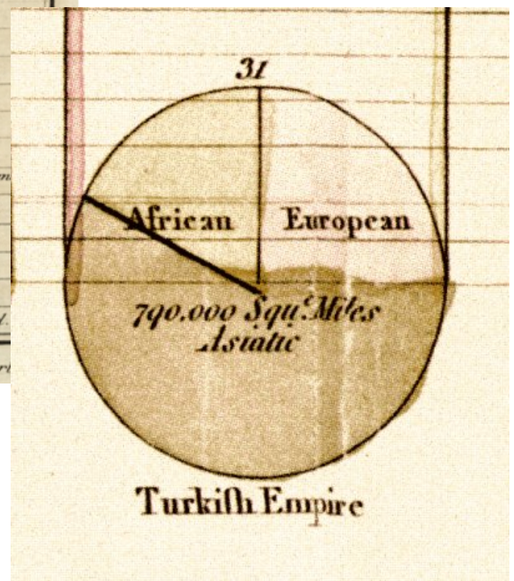


1786



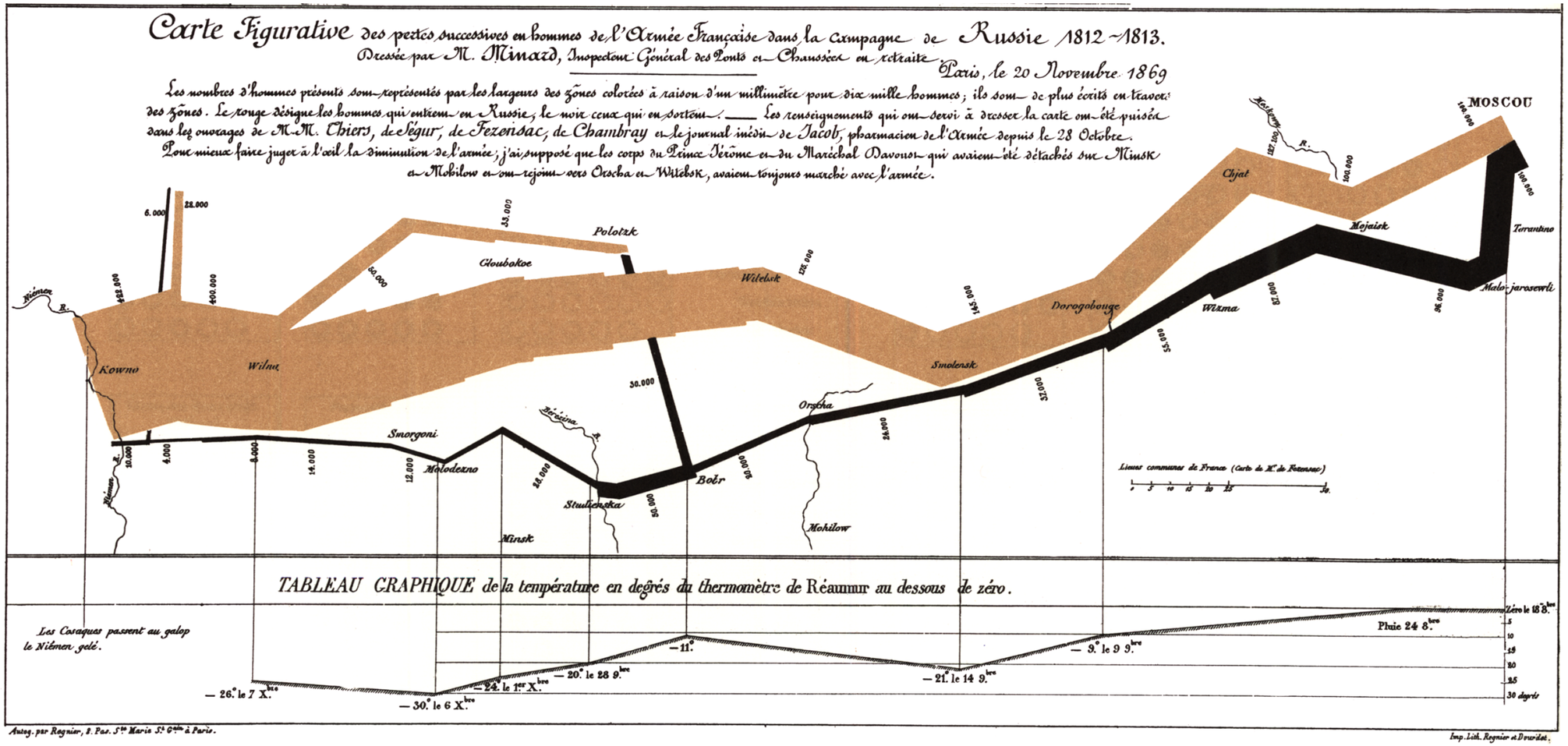
W. Playfair

1801



Defining data visualization (vis)

- **Visual representation of facts, events, elements and their attributes that are encoded by data**
- **Not necessarily connected with computer science: data visualization exists since centuries**



[M. Minard, 1869]

Defining data visualization in the age of computers

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

T. Munzner

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.

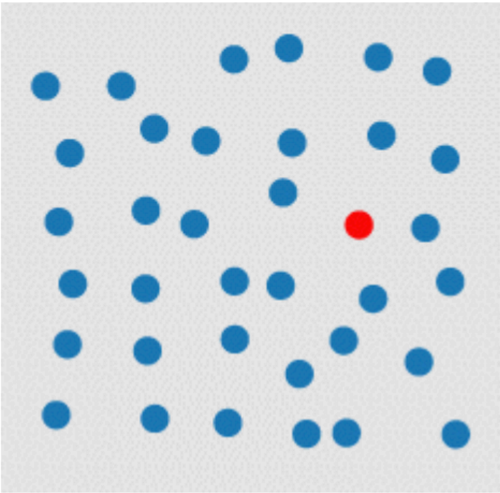
Visualization is suitable when there is a need to augment human capabilities rather than replace people with computational decision-making methods.

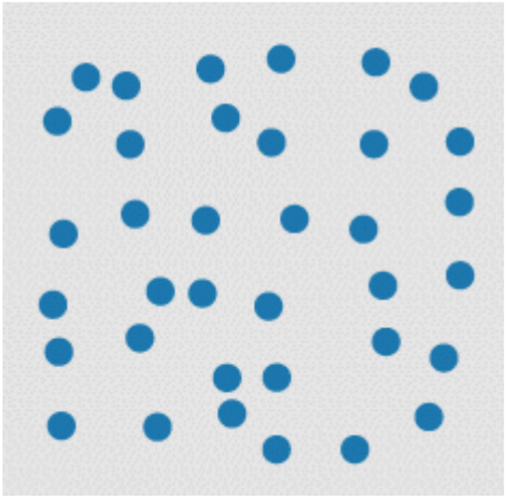
- don't need vis when fully automatic solution exists and is trusted
- many analysis problems ill-specified
  - don't know exactly what questions to ask in advance → need for **data exploration**
- possibilities
  - long-term use for end users (e.g. exploratory analysis of scientific data)
  - presentation 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 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*





When not to visualize?
When to automate?

- Well defined question on well-defined dataset
  - *Which gene is most frequently mutated in this set of patients?*
  - *What is the current unemployment rate?*
- Decisions needed in minimal time
  - *High frequency stock market trading: which stock to buy/sell?*
  - *Manufacturing: is bottle in production line broken?*

Why show the data in detail?
Why not rely on statistics?

- data may be many
- statistics help summarising and finding trends

but…

- summaries lose information
- detail allows us to
  - confirm expected and find unexpected patterns
  - assess validity of statistical model

Why show the data in detail?

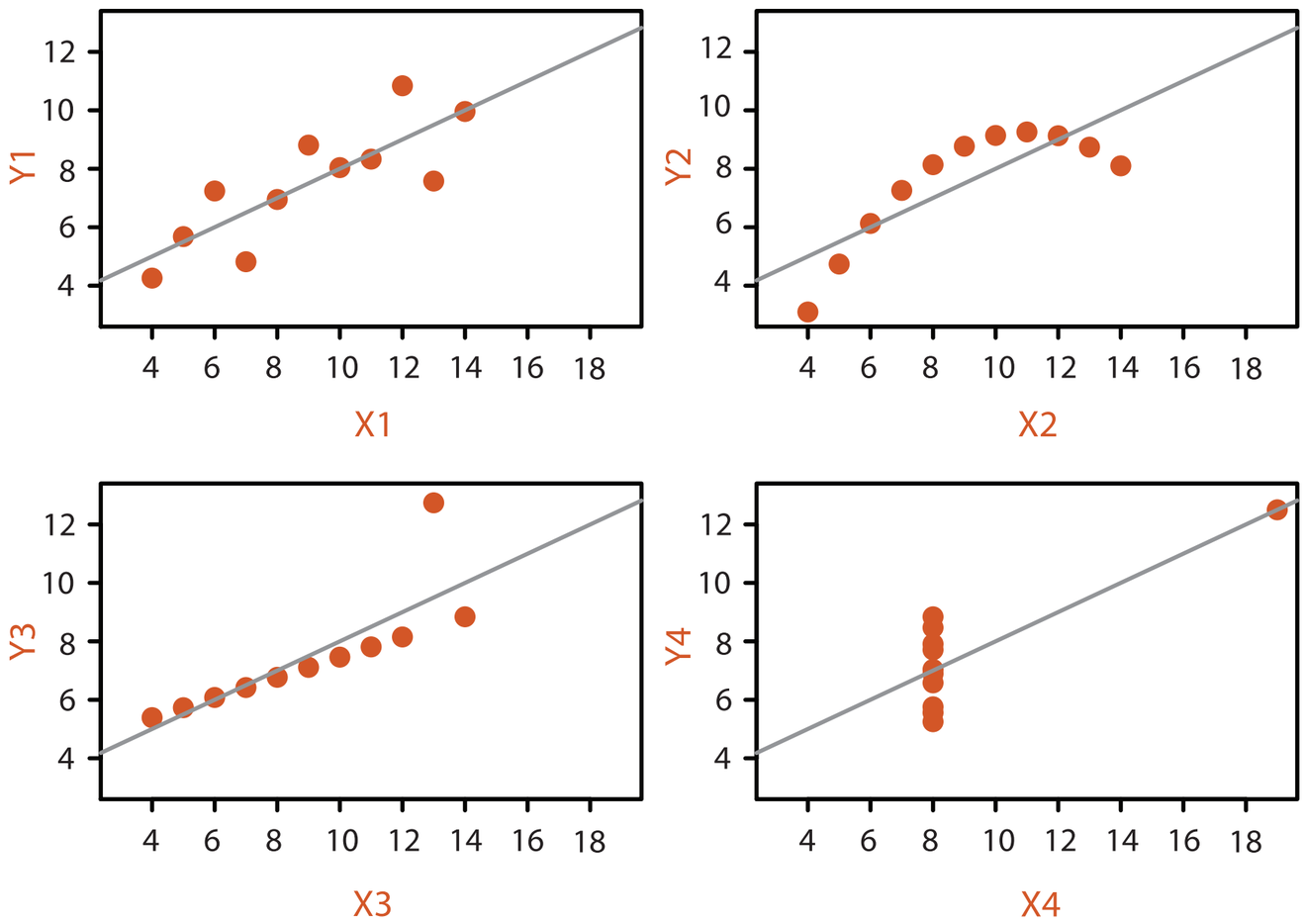
- Example: the Ancombe's quartet

I		II		III		IV	
x	y	x	y	x	y	x	y
10.0	8.04	10.0	9.14	10.0	7.46	8.0	6.58
8.0	6.95	8.0	8.14	8.0	6.77	8.0	5.76
13.0	7.58	13.0	8.74	13.0	12.74	8.0	7.71
9.0	8.81	9.0	8.77	9.0	7.11	8.0	8.84
11.0	8.33	11.0	9.26	11.0	7.81	8.0	8.47
14.0	9.96	14.0	8.10	14.0	8.84	8.0	7.04
6.0	7.24	6.0	6.13	6.0	6.08	8.0	5.25
4.0	4.26	4.0	3.10	4.0	5.39	19.0	12.50
12.0	10.84	12.0	9.13	12.0	8.15	8.0	5.56
7.0	4.82	7.0	7.26	7.0	6.42	8.0	7.91
5.0	5.68	5.0	4.74	5.0	5.73	8.0	6.89

Identical statistics	
x mean	9
x variance	10
y mean	7,5
y variance	3,75
x/y correlation	0,816

Why show the data in detail?

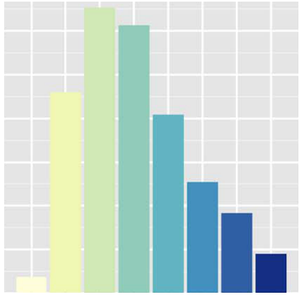
- Datasets may represent very different situations!
  - confirm expected and find unexpected patterns
  - assess validity of statistical model



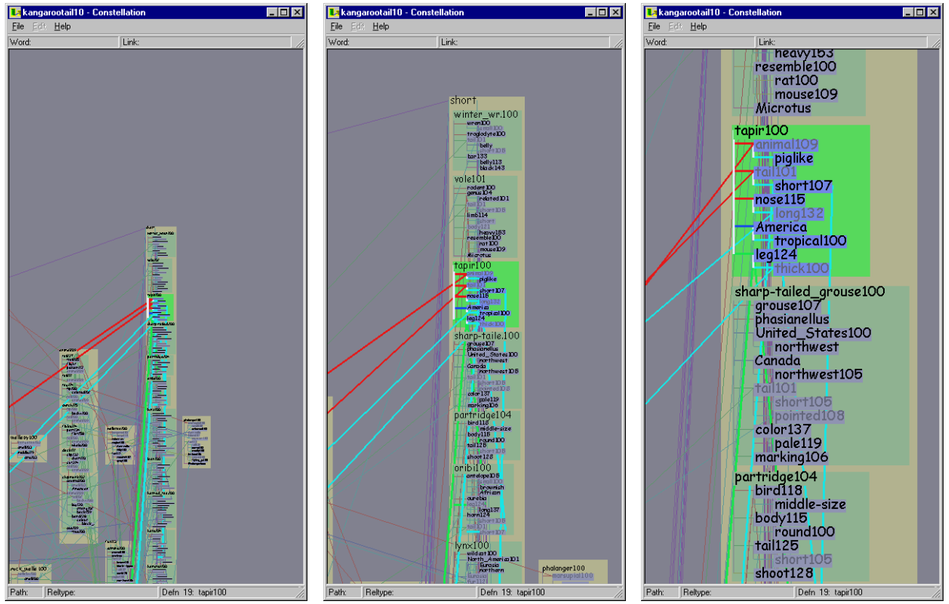
Idiom design space

The design space of possible vis idioms is huge, and includes the considerations of both how to create and how to interact with visual representations.

- **idiom**: distinct approach to creating or manipulating visual representation
  - how to draw it: **visual encoding** idiom
    - many possibilities for how to create



  - how to manipulate it: **interaction** idiom
    - even more possibilities
      - make single idiom dynamic
      - link multiple idioms together through interaction



[A layered grammar of graphics. Wickham. Journal of Computational and Graphical Statistics 19:1 (2010), 3–28.]

[Interactive Visualization of Large Graphs and Networks. Munzner. Ph.D. thesis, Stanford University Department of Computer Science, 2000.]

Why focus on tasks and effectiveness?

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

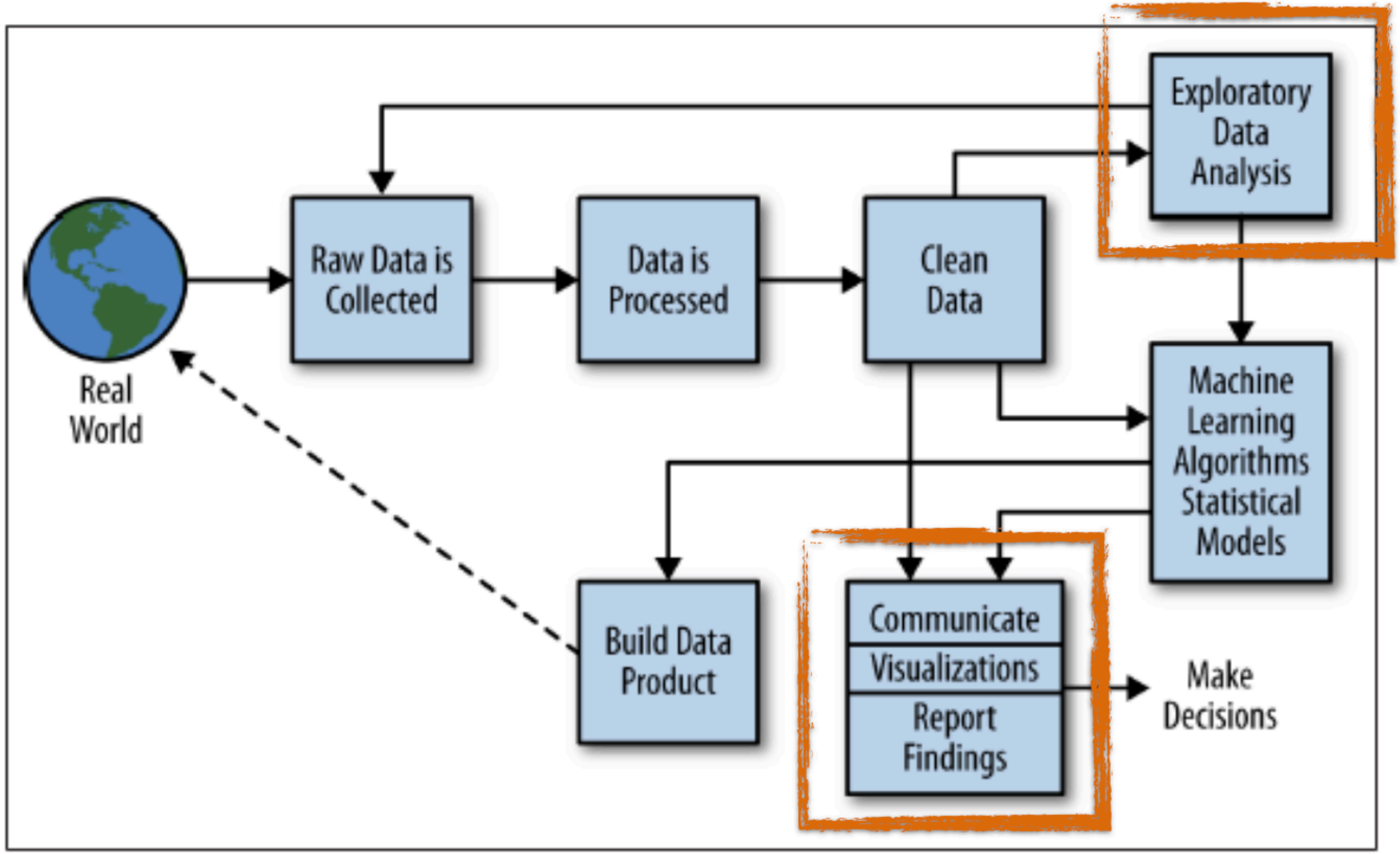
- tasks serve as constraint on design (as does data)
  - idioms do not serve all tasks equally!
  - challenge: recast tasks from domain-specific vocabulary to abstract forms
- most possibilities are ineffective
  - validation is necessary, but tricky
  - increases chance of finding good solutions if you understand full space of possibilities
- quest for effectiveness
  - find best combination of idioms to support your data and task
  - enable new kinds of analysis
  - speed up existing workflows

Resource limitations

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

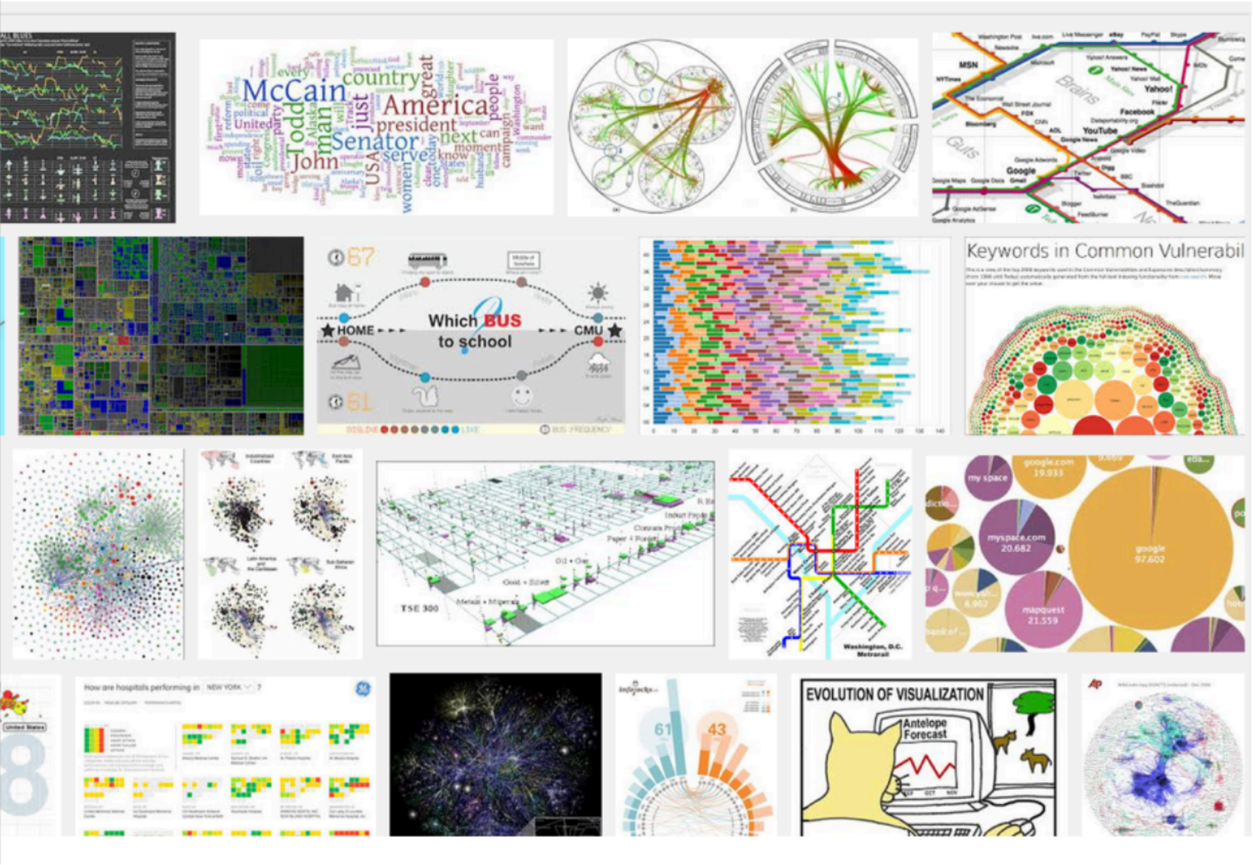
Visualization in the data science process



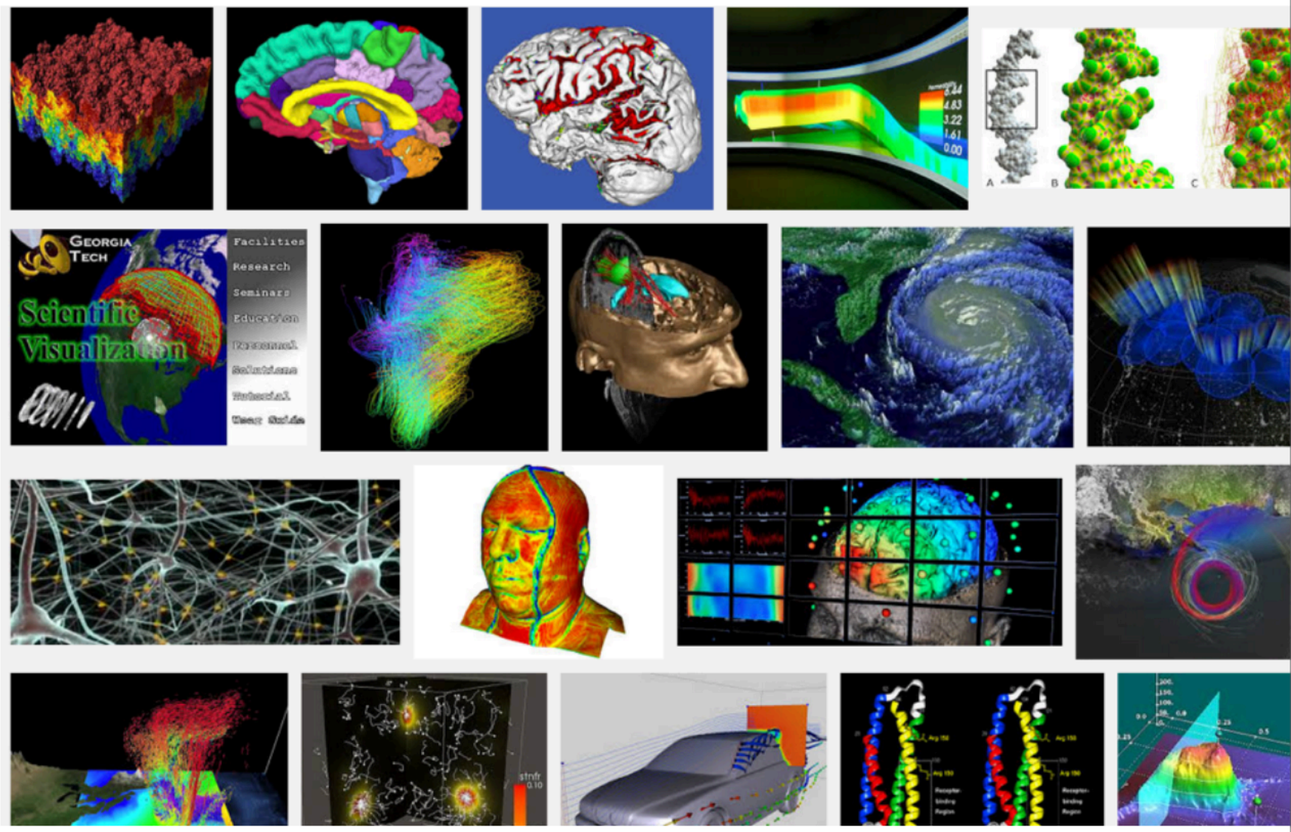
Data vis subdomains

- Information visualization (infovis):
  - abstract data
  - free to choose spatial layout
  - find best way to map data to space
- Scientific visualization (scivis):
  - spatial data
  - not free to choose spatial layout
  - find best way to depict reality
- Visual analytics:
  - Combination of data vis with statistics and machine learning
  - Focused on specific applications

Data vis subdomains



Infovis



Scivis

Data vis subdomains

- Information visualization (infovis):
  - abstract data
  - free to choose spatial layout
  - find best way to map data to space

In this course we focus here

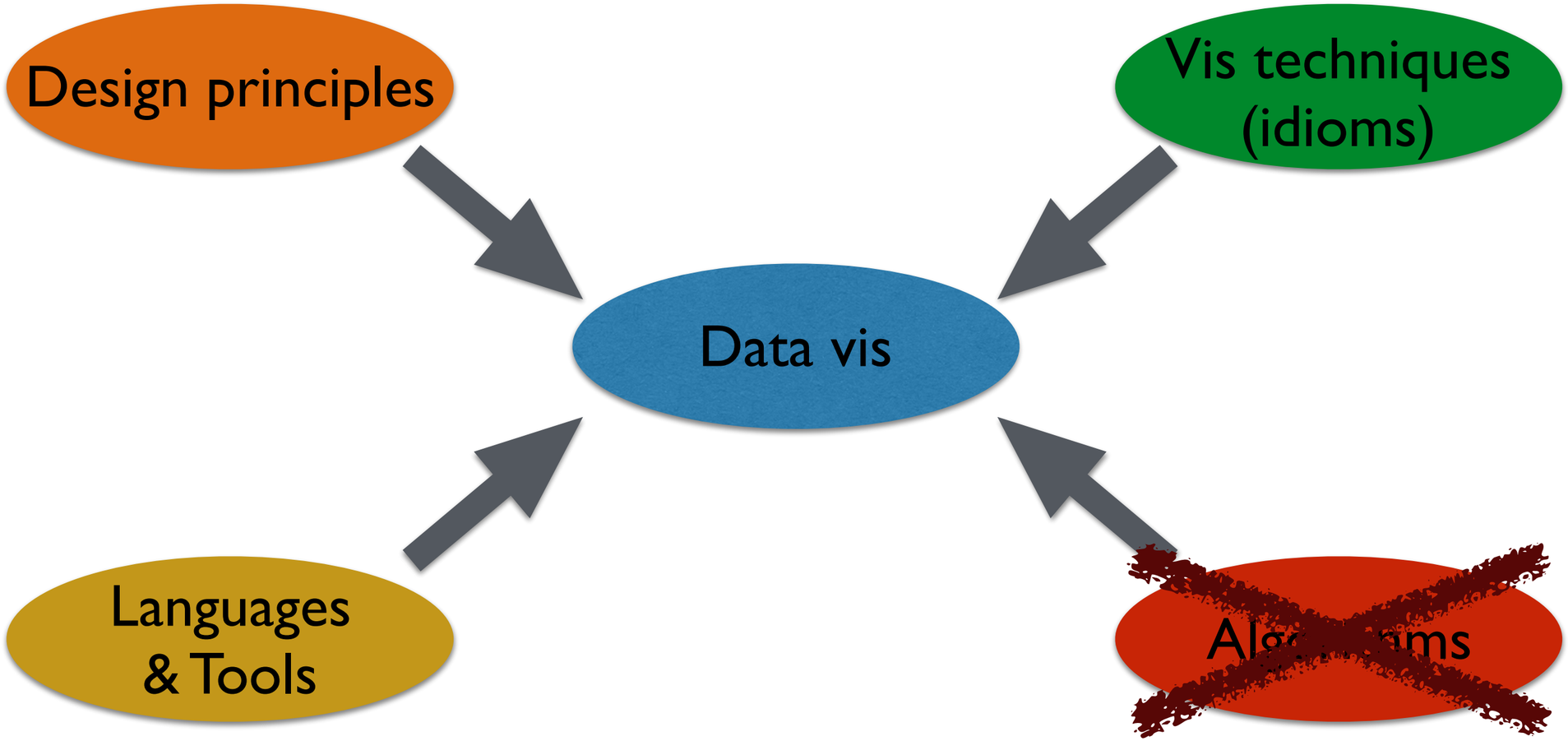
- Scientific visualization (scivis):
  - spatial data
  - not free to choose spatial layout
  - find best way to depict reality
- Visual analytics:
  - Combination of data vis with statistics and machine learning
  - Focused on specific applications

Data Vis vs. Computer Graphics

- Computer Graphics more general:
  - the field of study used to create images/animations with a computer
  - devoted to 3D for a large fraction (but also many 2D techniques)
    - digital modeling of 3D entities
    - from models to synthetic images
    - *very* extended field

  Follow course on Computer Graphics and Augmented Reality!
- Data Vis relies on computer graphics:
  - Infovis mostly built on 2D techniques (3D in Infovis usually bad)
  - Scivis uses 2D, 3D, time and projections from higher dimensions

The basic ingredients



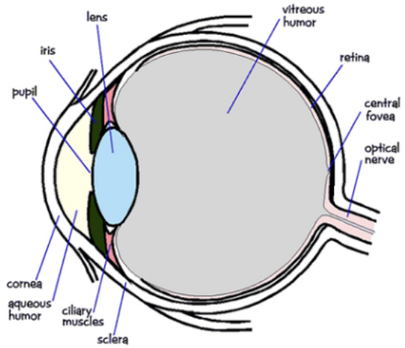
Statistics
Machine learning
Data mining
Graph analysis
Geometry processing

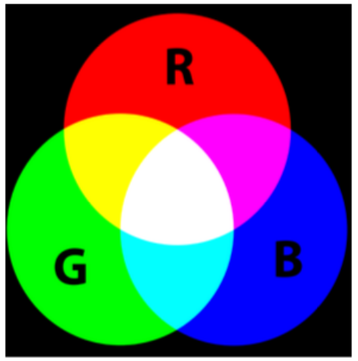
Seen in other courses

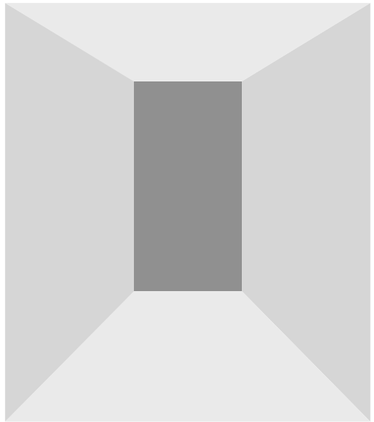
Part I - Design principles

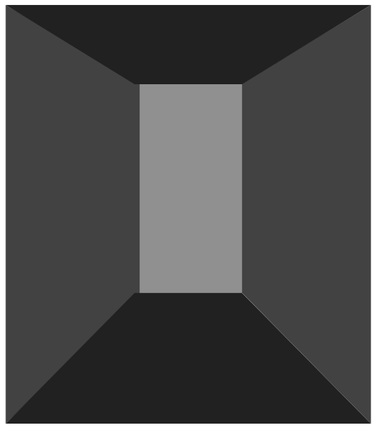
Perception, cognition, color

- Perception vs Cognition
- How does human vision work?
- How much can we trust it?
- Color:
  - how to encode it
  - how to use it in vis



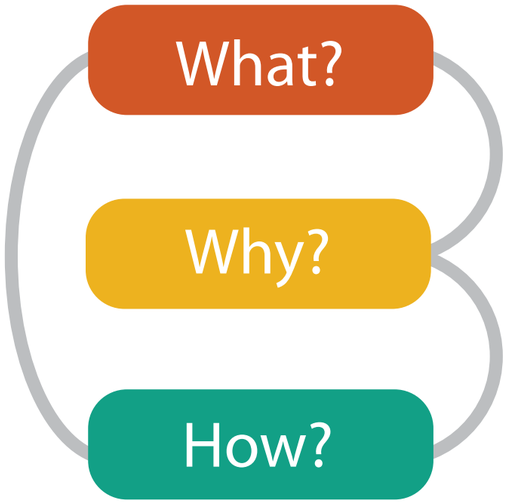






Analysis: What, why, and how

- what is shown?
  - **data** abstraction
- why is the user looking at it?
  - **task** abstraction
- how is it shown?
  - **idiom**: visual encoding and interaction



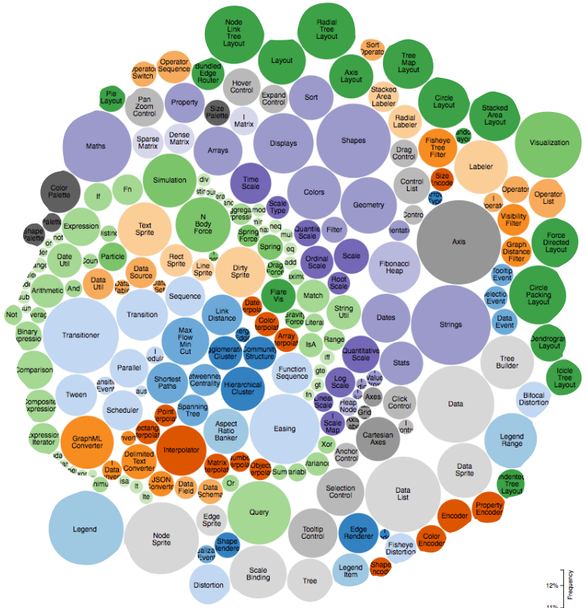
- abstract vocabulary avoids domain-specific terms
  - translation process iterative, tricky
- what-why-how analysis framework as scaffold to think systematically about design space

Interaction

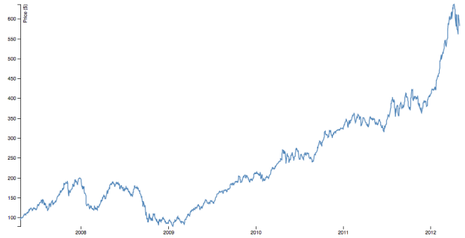
- Single view / Multiple views / Linked views
- Change over time / Transitions / Storytelling
- Navigating the vis space
- Semantic zooming
- Focus + Context (aggregation, reduction, layering, embedding)
- ...

Part II - Vis techniques

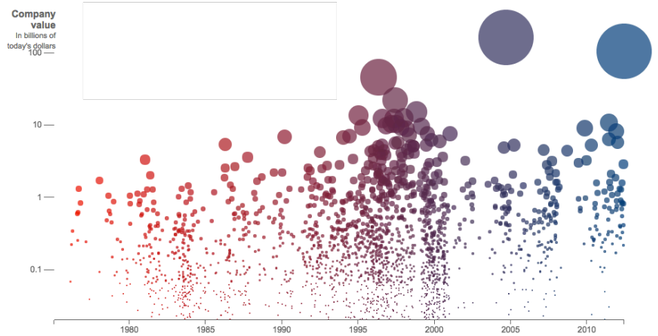
Idioms - basic



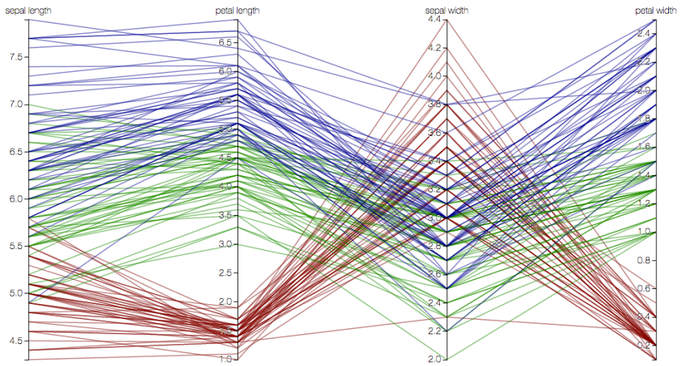
Bubble pack



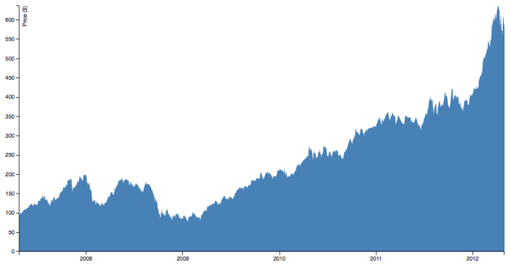
Line chart



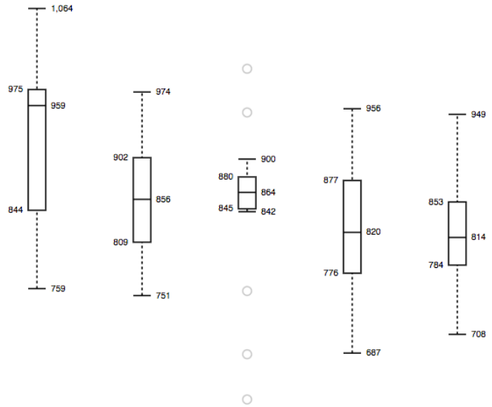
Scatterplot



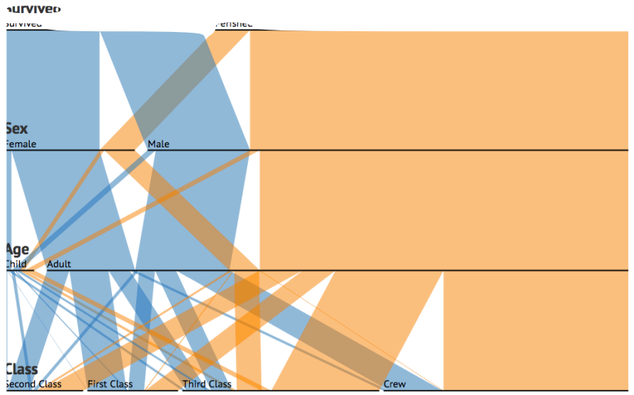
Parallel coordinates



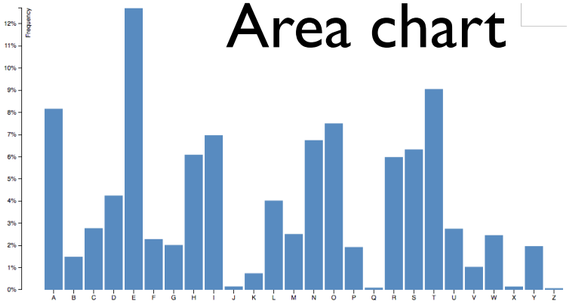
Area chart



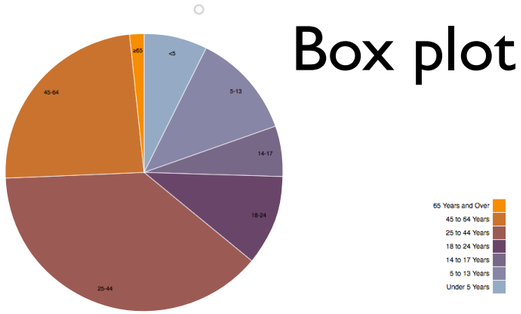
Box plot



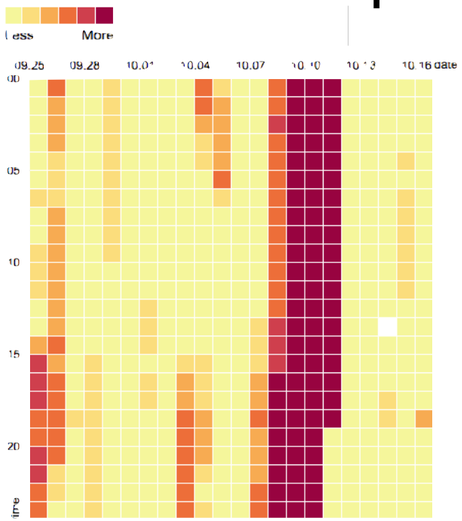
Parallel sets



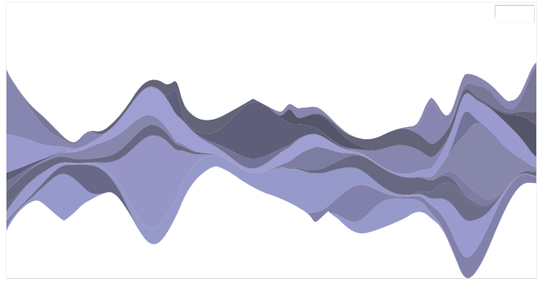
Bar chart



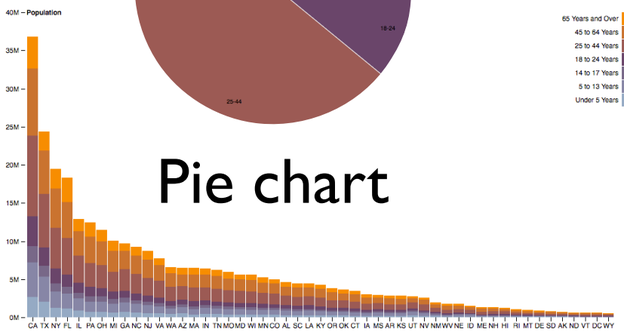
Pie chart



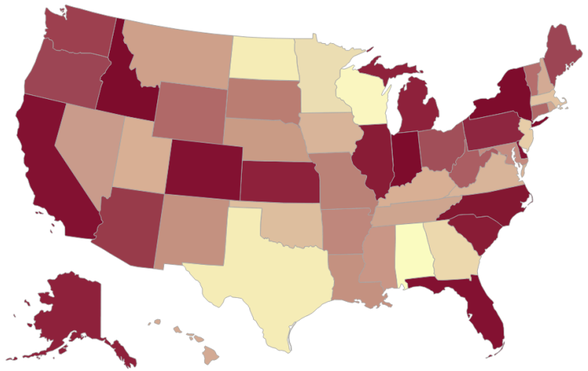
Heat map



Streamlines

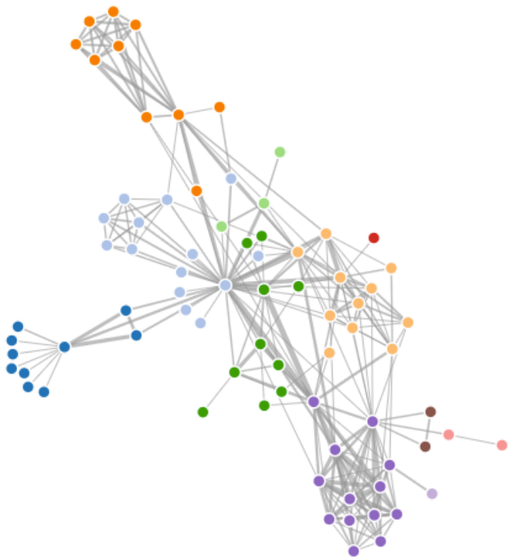


Stacked bar chart

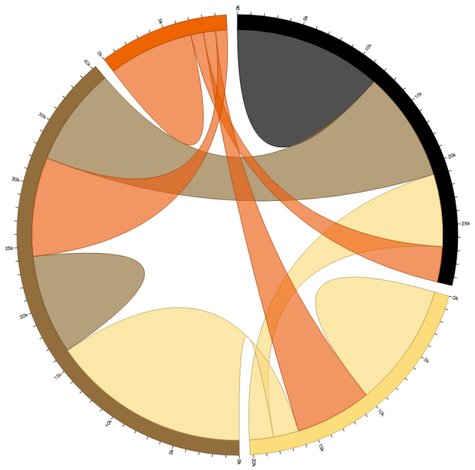


Choropleth map

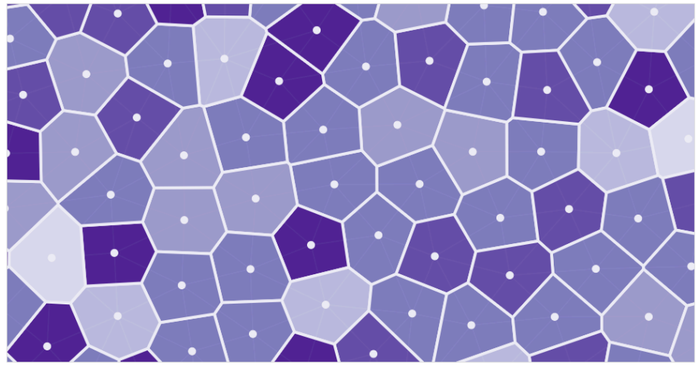
Idioms - networks and trees



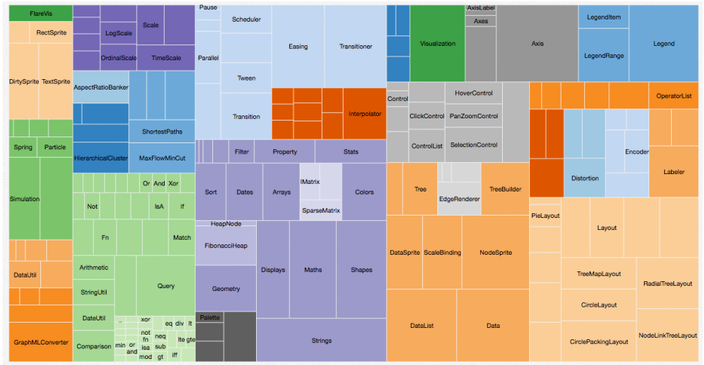
Force-directed graph



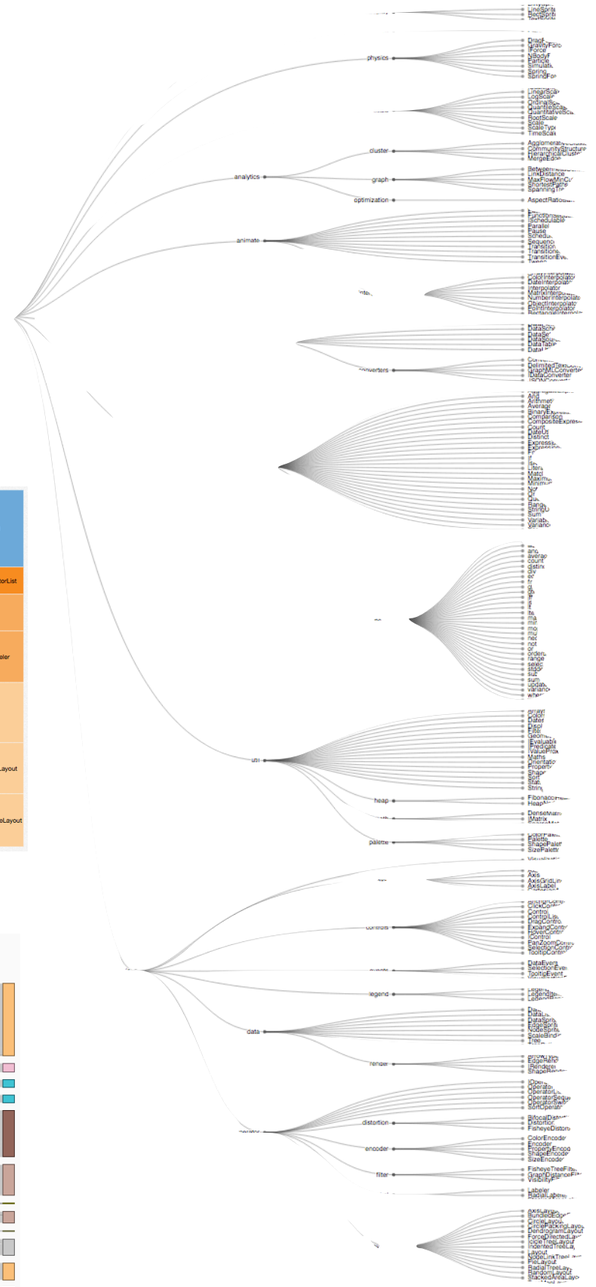
Chord diagram



Voronoi diagram

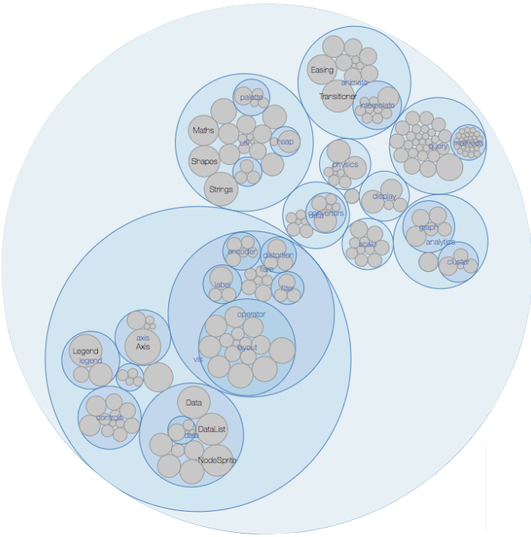


Treemap

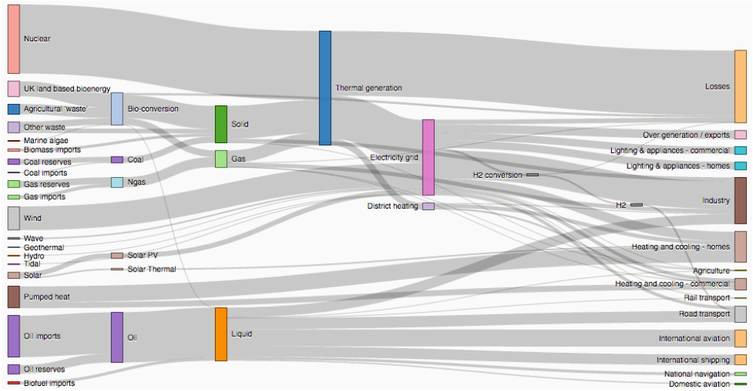


Dendogram

Radial tree

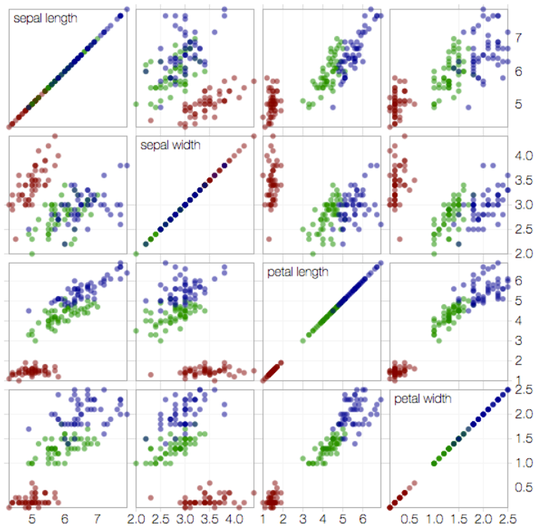


Hierarchical circle packing

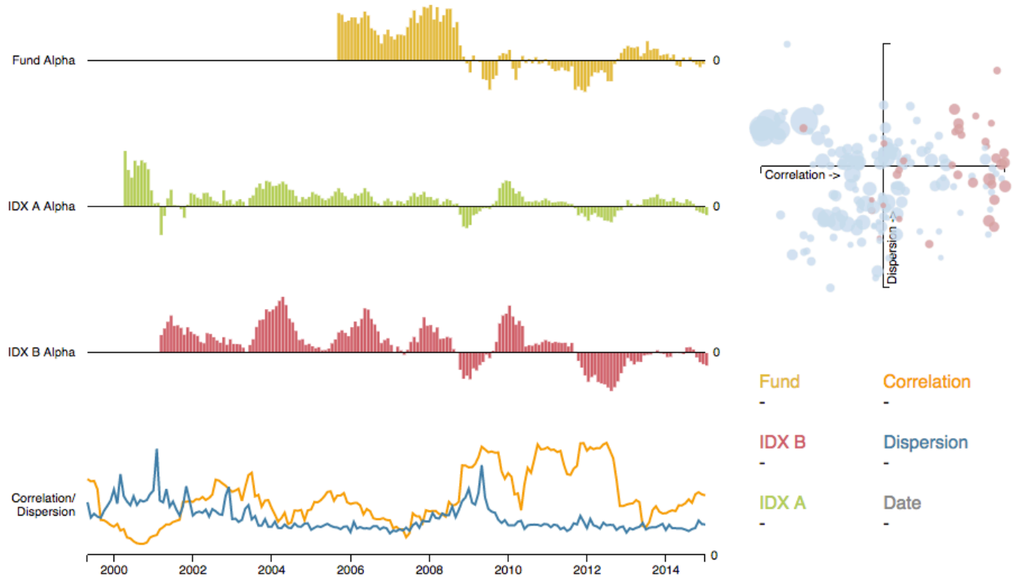


Sankey diagram

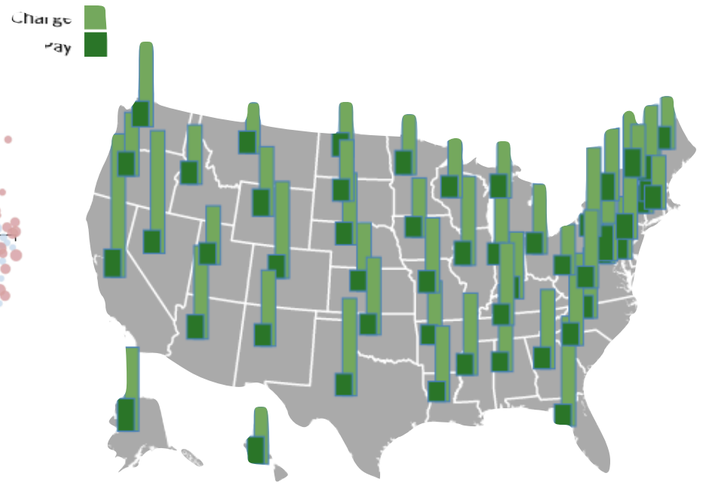
Idioms - composite



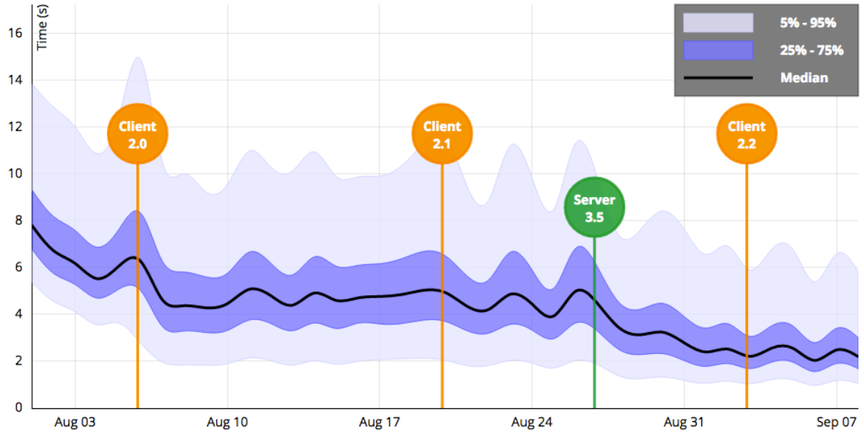
Scatterplot matrix



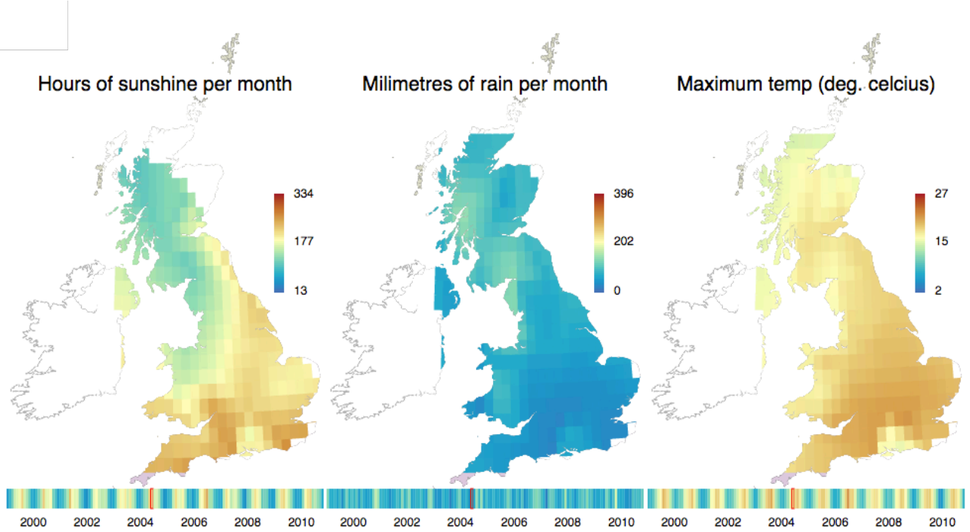
Bar & line & scatter



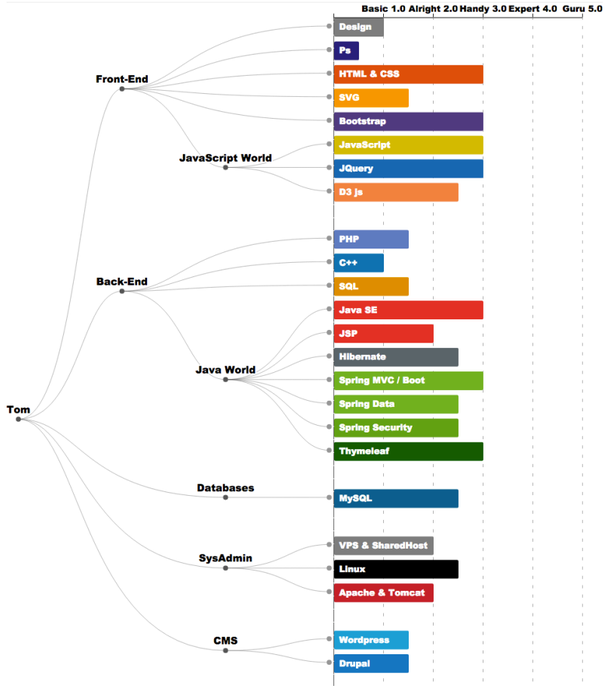
Map & bar chart



Trendchart (area+line)



Map & heat map



Dendogram & bar

Part III - Languages & Tools

Alternatives

- Low-level languages:
  - Computer graphics API's (OpenGL, DirectX)
  - Maximum flexibility, steep learning curve
  - Probably too much power for vis… and everything must be made from scratch
- Mid-level libraries:
  - Designed to support Vis (JS D3, Python Pandas/Matplotlib/Seaborn or Bokeh)
  - Rather flexible, more gentle learning curve, programming
- Toolkits:
  - Designed for professionals, non-CS people (Tableau)
  - More limited and rigid, easy to learn

D3 - Data Driven Documents

- Javascript library to link data do HTML documents and do Vis
- declarative: just say **what** to do, not **how** to do it
- Requires knowledge of HTML, CSS, SVG, Javascript (and the DOM)
- pros
  - seamless interoperability with Web
  - explicit transforms of scene with dependency info
  - massive user community, many thirdparty apps/libraries on top of it, lots of docs
- cons
  - different from traditional programming model
- example apps: many

Next lectures

- We start with design principles:
  - perception, cognition, color
  - the What / Why / How abstraction
- As soon as you're aligned on HTML/CSS/Javascript (no more than 2 weeks to do it) we start with SVG and then D3 → Homework!
- Most of the rest will be concerned with idioms:
  - vis idioms
  - interaction idioms
  - how to combine them
  - strategies to manage complexity