NDBI040

Big Data Management and NoSQL Databases

Lecture 12. Visualization of (Big) Data

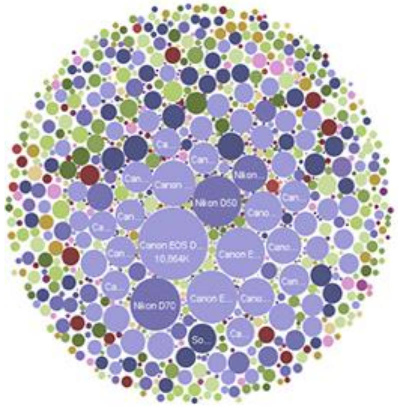
RNDr. David Hoksza, Ph.D.

Doc. RNDr. Irena Holubova, Ph.D.

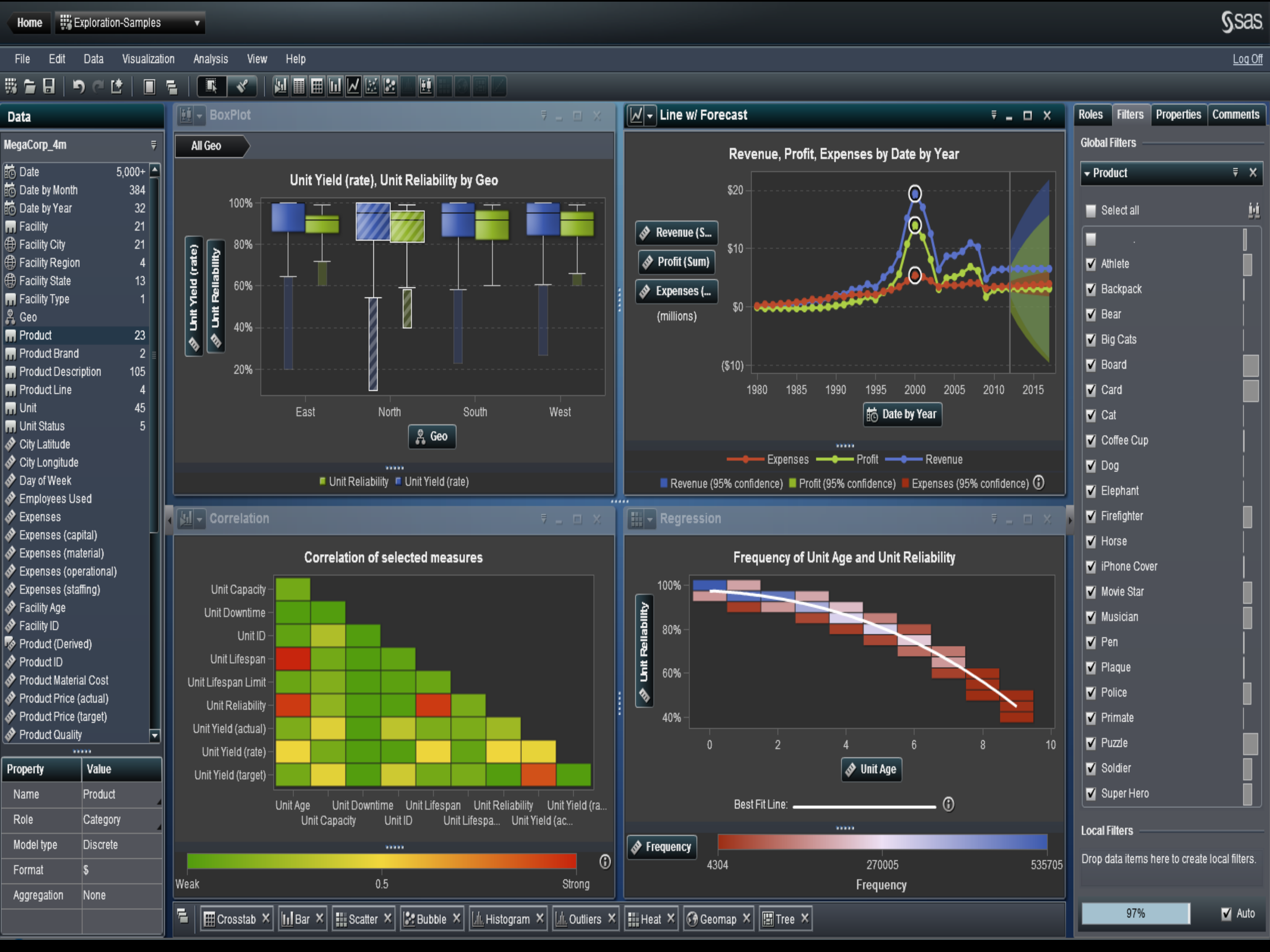
{hoksza,holubova}@ksi.mff.cuni.cz

http://www.ksi.mff.cuni.cz/~holubova/NDBI040/

Motivation for (Big) Data Visualization



- Data visualization = creation and studying of visual representation of data
  - □ Information abstracted in some schematic form
  - □ Including attributes, variables, …
- Purpose:
  - □ To communicate information clearly and effectively through graphical means
  - □ To help find the information needed more effectively and intuitively
- Both aesthetic form and functionality are required
- Even when data volumes are large, the patterns can be spotted quite easily (with the right data processing and visualization)
  - □ Simplification of Big Data management
  - □ Picking up things with the naked eye that would otherwise be hidden



Date	5,000+
Date by Month	384
Date by Year	32
Facility	21
Facility City	21
Facility Region	4
Facility State	13
Facility Type	1
Geo	
Product	23
Product Brand	2
Product Description	105
Product Line	4
Unit	45
Unit Status	5
City Latitude	
City Longitude	
Day of Week	
Employees Used	
Expenses	
Expenses (capital)	
Expenses (material)	
Expenses (operational)	
Expenses (staffing)	
Facility Age	
Facility ID	
Product (Derived)	
Product ID	
Product Material Cost	
Product Price (actual)	
Product Price (target)	
Product Quality	

Property	Value
Name	Product
Role	Category
Model type	Discrete
Format	$
Aggregation	None

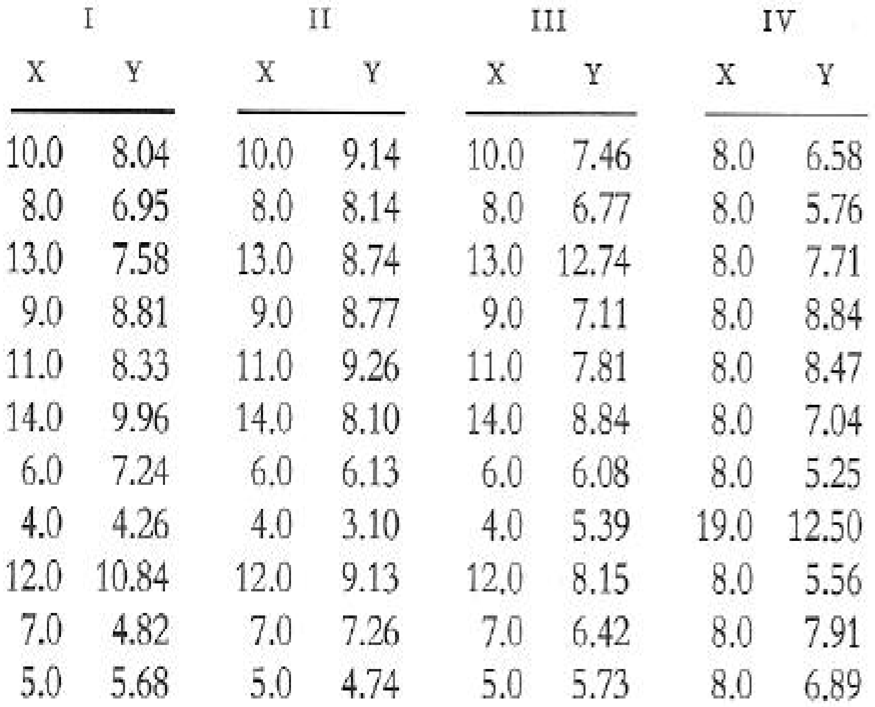
□ Select all
□
☑ Athlete
☑ Backpack
☑ Bear
☑ Big Cats
☑ Board
☑ Card
☑ Cat
☑ Coffee Cup
☑ Dog
☑ Elephant
☑ Firefighter
☑ Horse
☑ iPhone Cover
☑ Movie Star
☑ Musician
☑ Pen
☑ Plaque
☑ Police
☑ Primate
☑ Puzzle
☑ Soldier
☑ Super Hero

Local Filters

Drop data items here to create local filters.

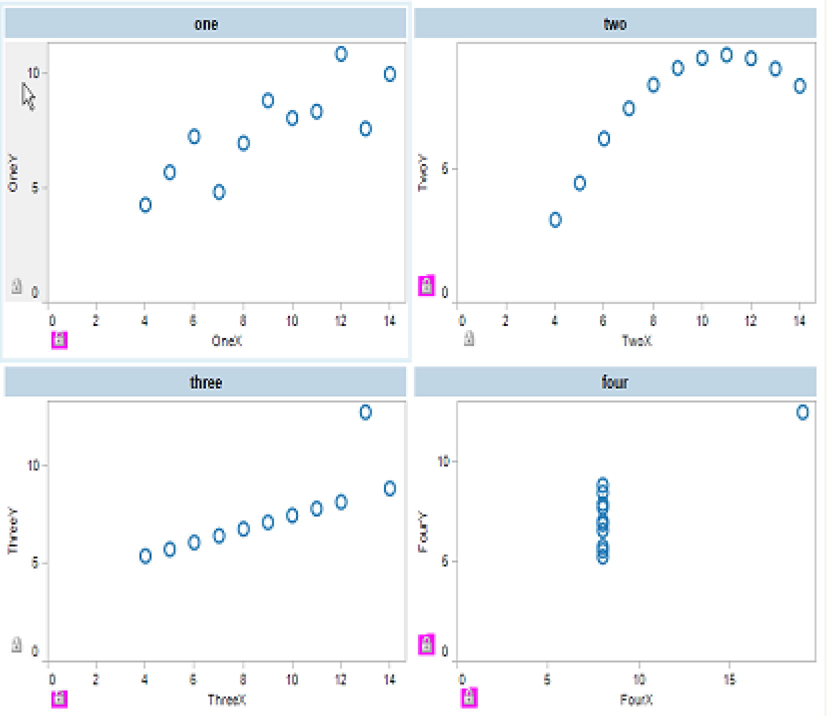
Motivation

Similar motivation as for statistics but visualization can reveal/distinguish data/trends/patters, … which statistics can not



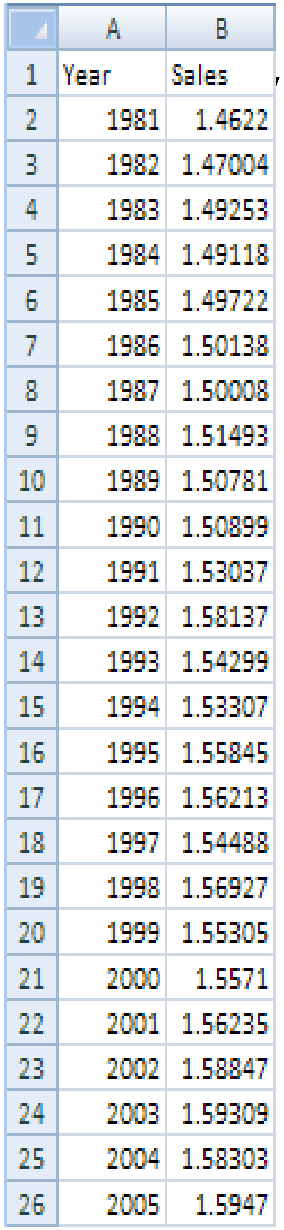
I		II		III		IV	
X	Y	X	Y	X	Y	X	Y
10.0	8.04	10.0	9.14	10.0	7.46	8.0	6.58
8.0	6.95	8.0	8.14	8.0	6.77	8.0	5.76
13.0	7.58	13.0	8.74	13.0	12.74	8.0	7.71
9.0	8.81	9.0	8.77	9.0	7.11	8.0	8.84
11.0	8.33	11.0	9.26	11.0	7.81	8.0	8.47
14.0	9.96	14.0	8.10	14.0	8.84	8.0	7.04
6.0	7.24	6.0	6.13	6.0	6.08	8.0	5.25
4.0	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

Source: Tufte, Edward R (1983), *The Visual Display of Quantitative Information,* Graphics Press

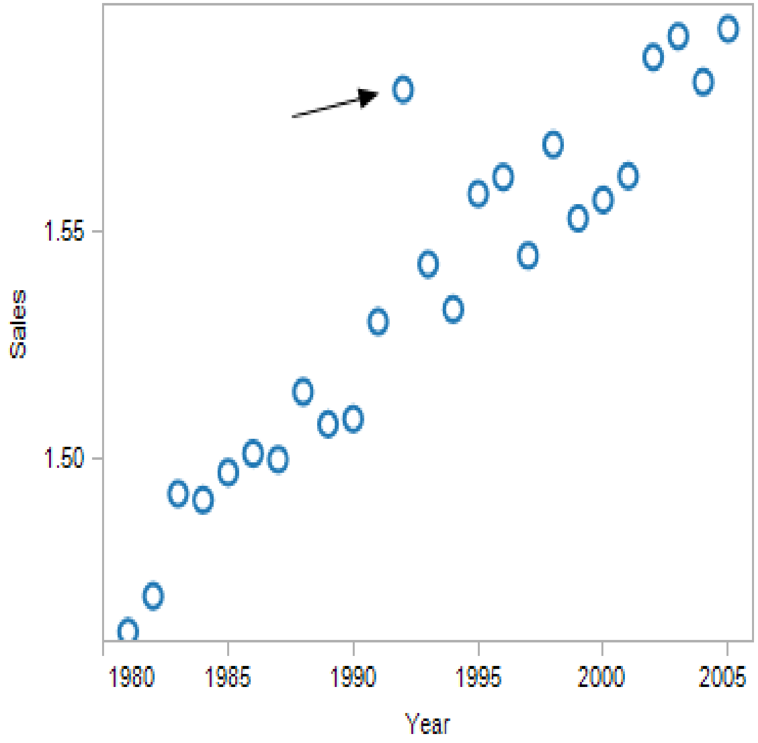


Four data sets with nearly identical linear model (mean, variance, linear regression line, …)

ation



	A	B
1	Year	Sales
2	1981	1.4622
3	1982	1.47004
4	1983	1.49253
5	1984	1.49118
6	1985	1.49722
7	1986	1.50138
8	1987	1.50008
9	1988	1.51493
10	1989	1.50781
11	1990	1.50899
12	1991	1.53037
13	1992	1.58137
14	1993	1.54299
15	1994	1.53307
16	1995	1.55845
17	1996	1.56213
18	1997	1.54488
19	1998	1.56927
20	1999	1.55305
21	2000	1.5571
22	2001	1.56235
23	2002	1.58847
24	2003	1.59309
25	2004	1.58303
26	2005	1.5947



Find an outlier….

Data Visualization

- Information visualization has two equally important aspects
  - Structural modeling
    - Detection, extraction and simplification of the underlying information
  - Graphical representation
    - Transform initial representation into a graphical one which provides visualization of the structure
      - Different types of structures require different type of visualization
      - e.g., time series vs. hierarchical information

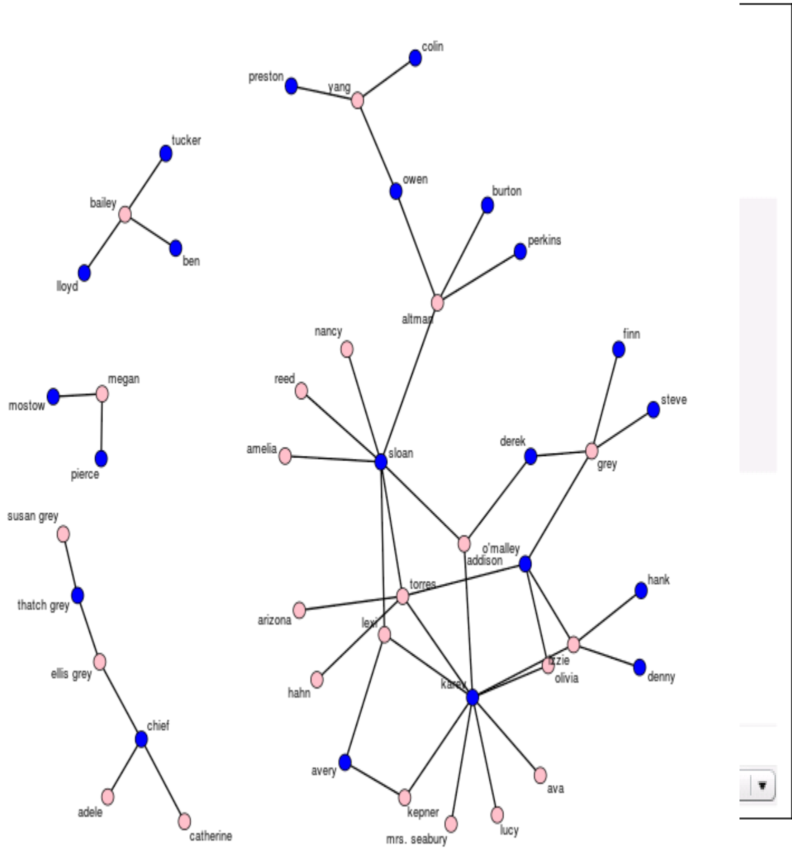
Big Data Visualization

- Decision about what technique to use became more difficult with Big Data
  - Visualization is needed to decide which portion of data to explore further
  - Visualization algorithms (i.e., graph drawing) should scale well to billions of entities (nodes)
    - The first application was probably the visualization of web-related data
      - i.e., pages, relations, traffic, …
  - New techniques may be needed
  - Trends might not be clear
  - Noise reduction might be even more necessary

Visualization Types

Data Relationships

- Scatter plot
  - Classical statistical diagram that lets us visualize relationships between numeric variables
  - Can carry additional information
- Matrix chart
  - Summarizes a multidimensional data set in a grid
- Network diagram
  - A set of objects (vertices) connected by edges
  - Visualization of the network is optimized to keep strongly related items in close proximity to each other



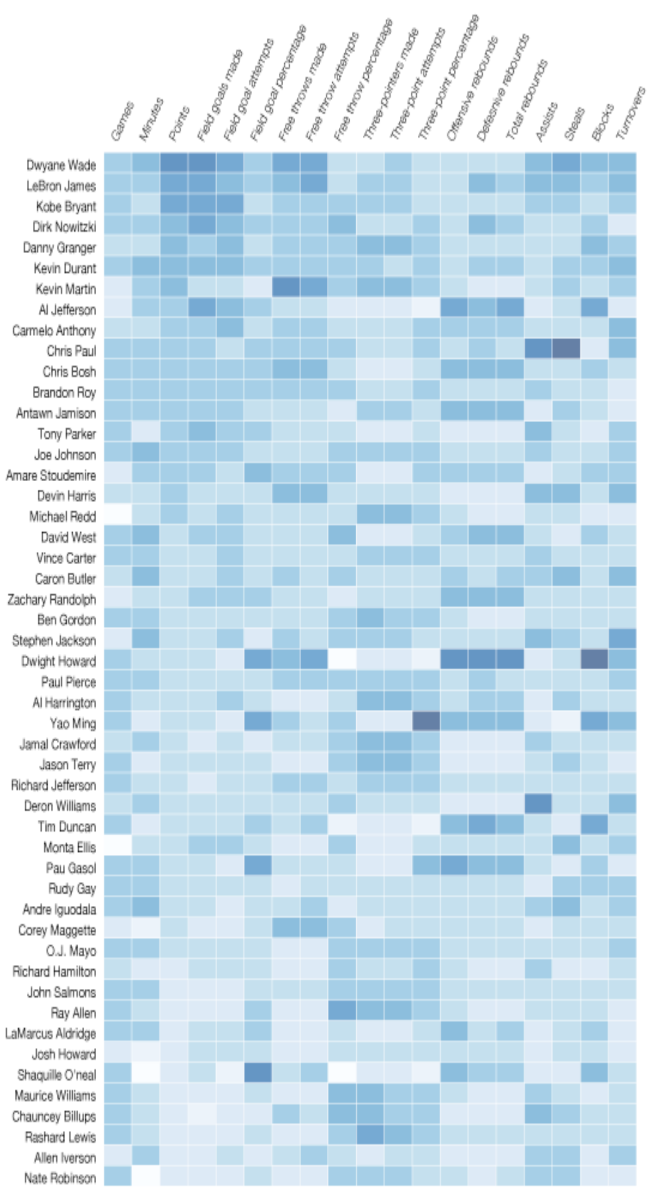
Visualization Types

Data Relationships

- Correlation matrix (heat map)
  - Combines data to quickly identify which variables are related
  - Shows how strong the relationship is between the variables

NBA per game performance of top 50 scorers

2008-2009 season

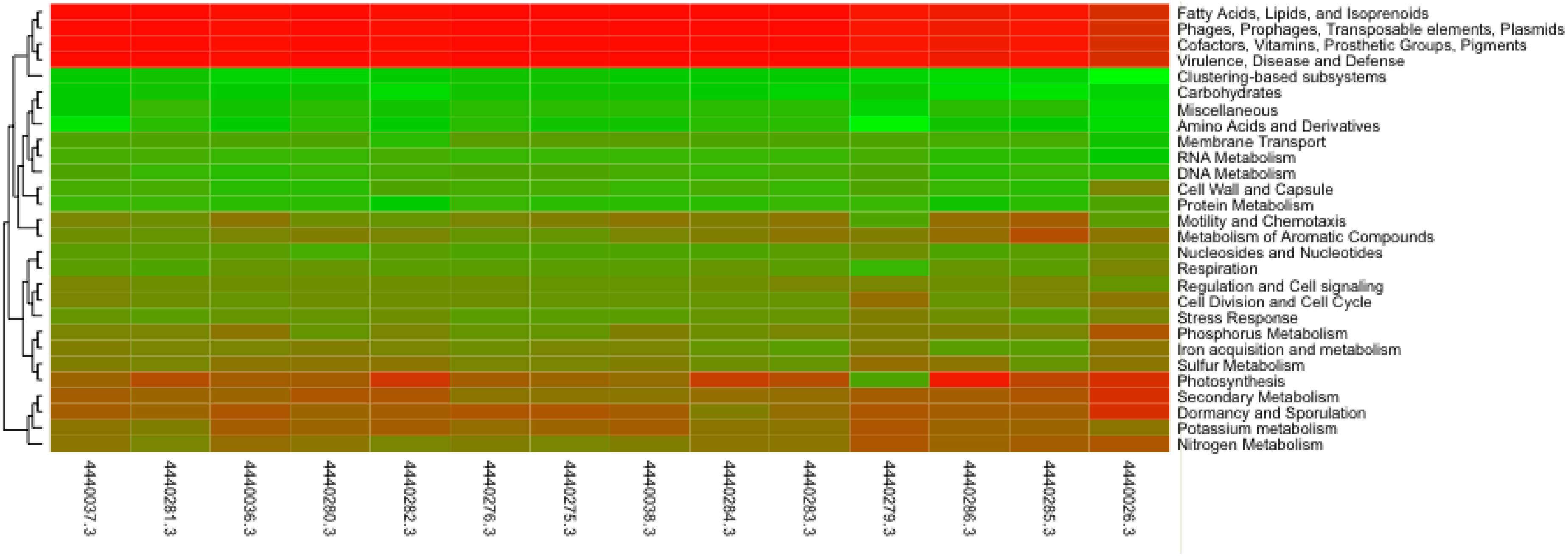


Source: databaseBasketball

Visualization Types

Data Relationships

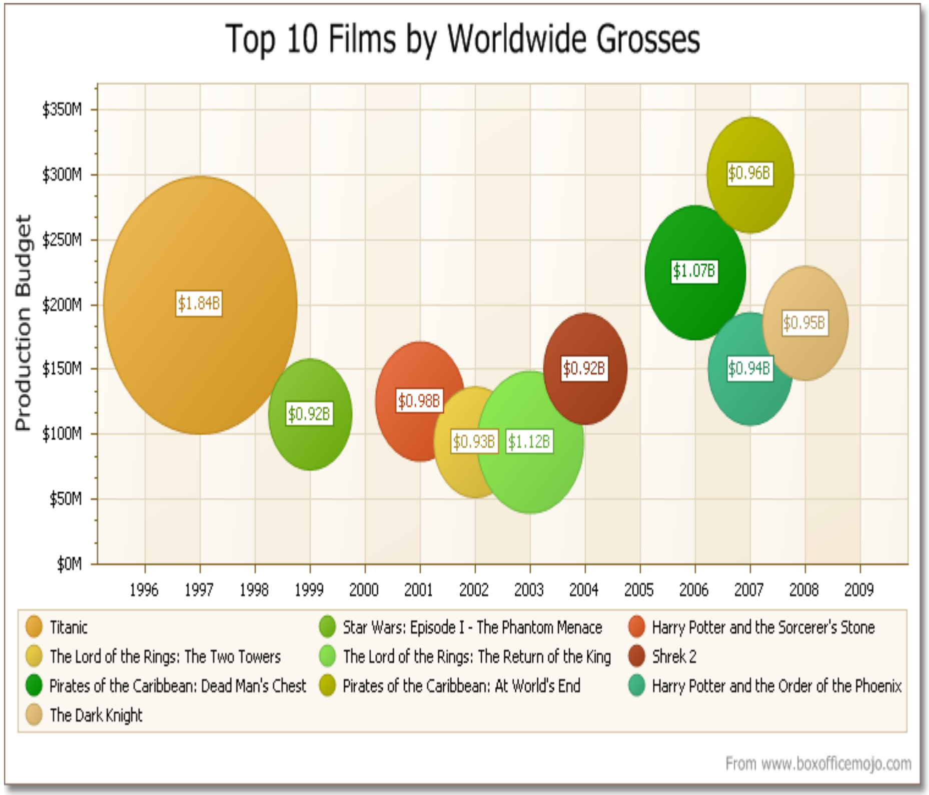
- Heat map is often combined with a dendrogram
  - Aggregates rows or columns based on their overall similarity into a tree structure



Visualization Types

Comparison of a Set of Values

- Bar Chart
  - □ Classical method for numerical comparisons
  - □ Histograms
  - □ Box plot (box-and-whisker plots)
    - Five statistics (minimum, lower quartile, median, upper quartile and maximum) summarizing the distribution of a set of data
- Bubble chart
  - □ Circles in a bubble chart represent different data values
    - The area of a circle corresponding to the value
  - □ The positions of the bubbles do not mean anything
    - Designed to pack the circles together with relatively little wasted space



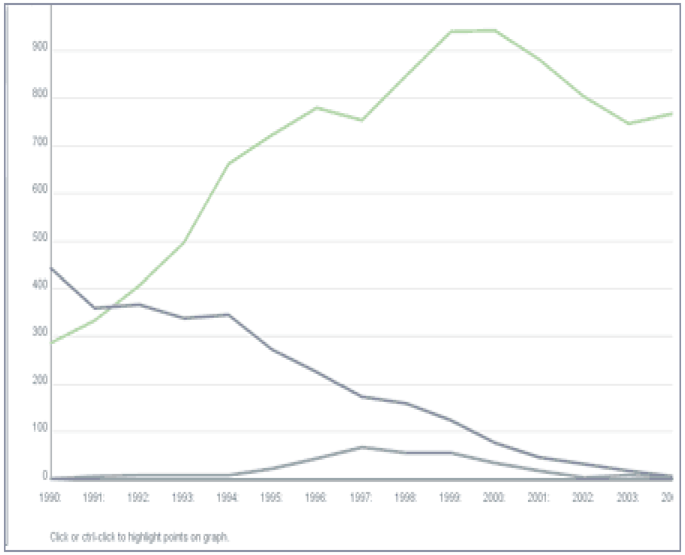
From www.boxofficemojo.com

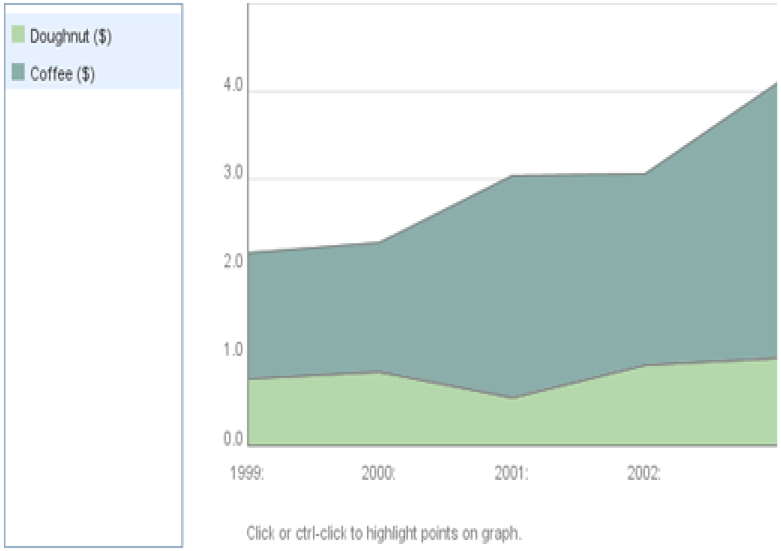
Experiment No.

Visualization Types

Trends over Time

- Line graph
  - Classical method for visualizing continuous change
- Stack graph
  - Visualizing change in a set of items
  - The sum of the values is as important as the individual items

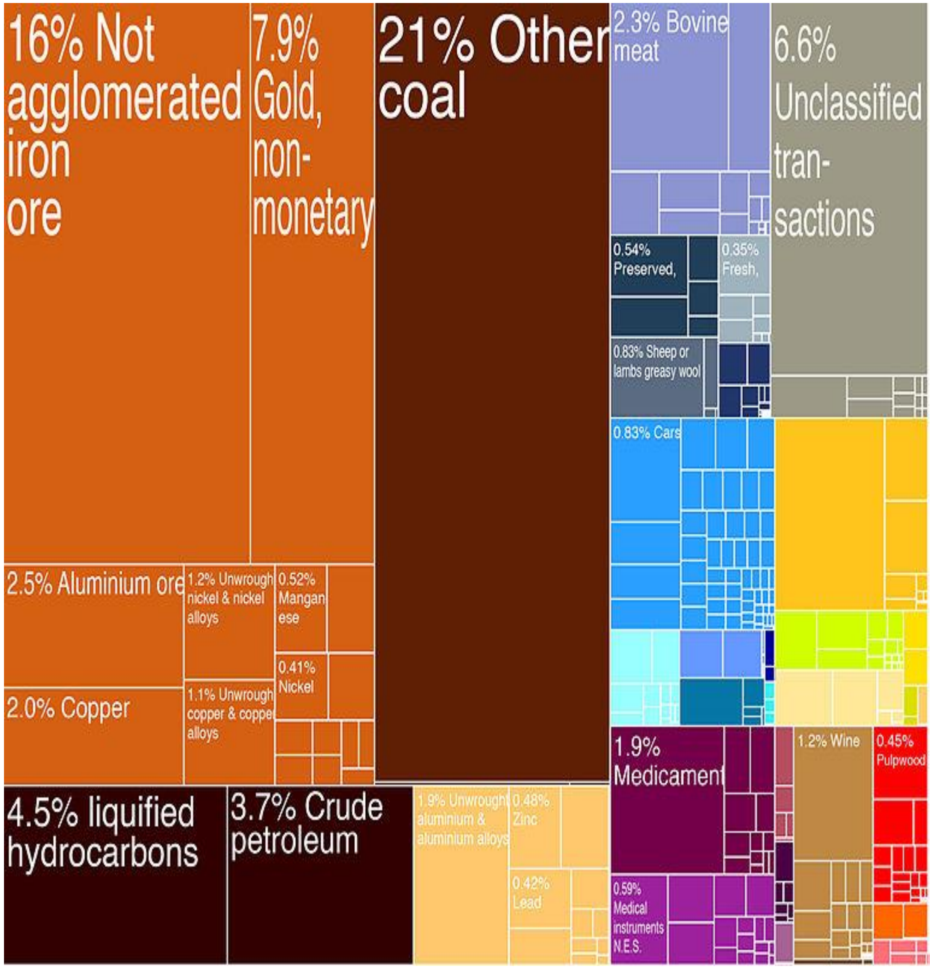




Visualization Types

Parts of a Whole

- Pie Chart
  - □ Percentages are encoded as "slices" of a pie, with the area corresponding to the percentage
- Treemap
  - □ Visualization of hierarchical structures
  - □ Effective in showing attributes of leaf nodes using size and color coding
  - □ Enable to compare nodes and sub-trees at varying depth



Economy of Australia

Visualization Types

Text Analysis

- Tag cloud
  - Visualization of word frequencies
    - i.e., how frequently words appear in a given text

BIG
DATA
ANALYTICS
STORAGE
TECHNOLOGIES
INFORMATION
INTERNET
SOFTWARE
MANAGEMENT
DATABASES
COMPLEX
NETWORKS
SYSTEMS
INCLUDE
SHARED
TIME
DISK
HUNDREDS
BIOLOGICAL
MASSIVELY
GARTNER
USING
WORKING
PERFORMANCE
GENOMICS
TOOLS
SET
WITHIN
PROCESS
DEFINITION
SEARCH
RECORDS
RELATIONAL
CAPACITY
PRACTITIONERS
TENS
USE
ONE
BUSINESS
MOVING
PARALLEL
SAN
TOLERABLE
SIZE
NEEDED
PETABYTES
PROCESSING
LOGS
DISTRIBUTED
CAPTURE
MANAGE
GROW
SOCIAL
LARGE
STORE
MAY
EVERY
CASE
LARGER
RESEARCH
TERABYTES
MPP
COST
CONTINUES
CITATION
DIFFICULTY
TARGET
ABILITY
SETS
EXAMPLES
APPLIED
ELAPSED
CURRENT
SENSOR
ARCHIVES

Which Visualization Technique to Use?

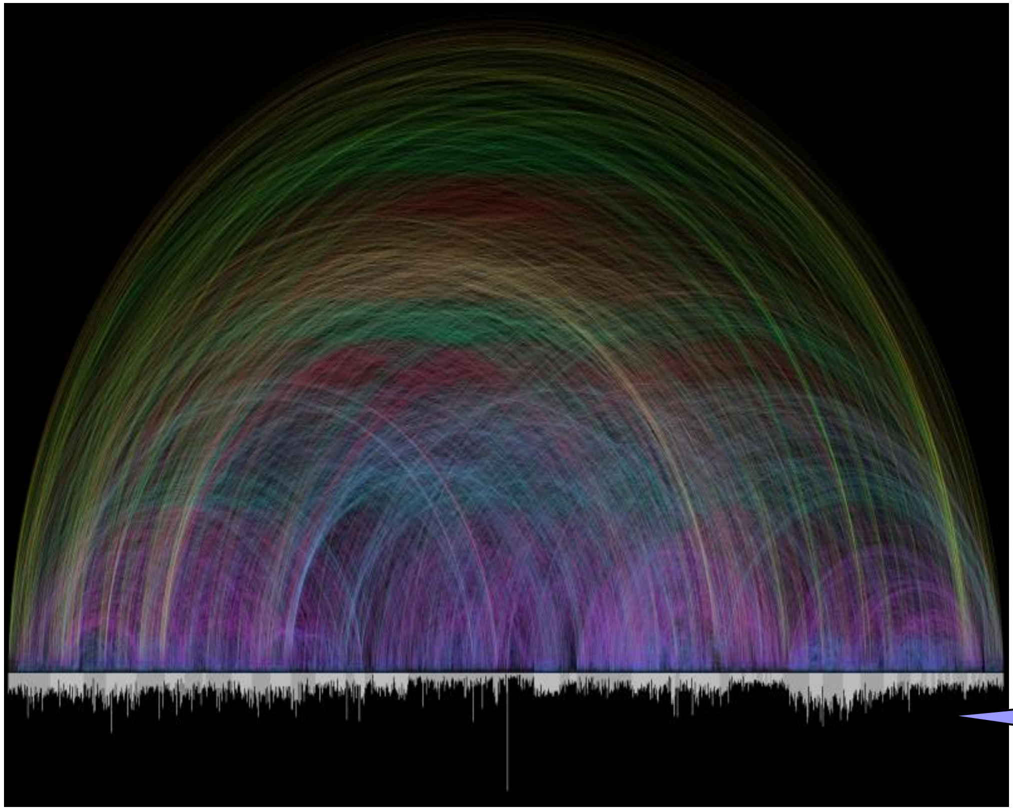
- New visualization software is capable of “guessing” the correct visualization based on the characteristics of the data
  - □ One-dimensional data ⇒ bar chart
  - □ Two-dimensional data ⇒ scatter plot
  - □ N-dimensional data ⇒ multiple scatter plots, matrix chart, …
  - □ Data with coordinates ⇒ map-based charts
- Offers options
- Trend: to simplify the process for common users

Big Data Visualization

- The goal of visualizing Big Data is usually to make sense of a large amount of interlinked information
- In interconnected data the connections between objects are difficult to organize on a linear layout
  - **Circular representations**
  - **Network diagrams**
- Typical “topologies” one can encounter (a bit confusing term based on Manuel Lima’s “Visual Complexity” – see references) include arc diagrams, centralized burst, centralized ring, globe, circular ties or radial convergence
  - And many more…

Arc Diagram

- Vertices are placed on a line and edges are drawn as semicircles
- Arcs represent relationships
  - Colors can encode, e.g., distance

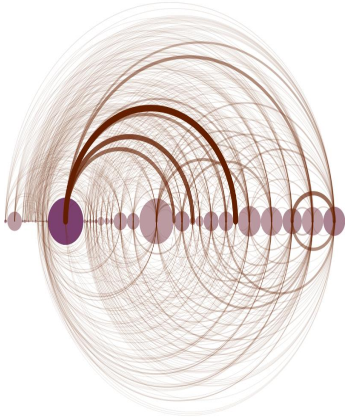


grey/white = book

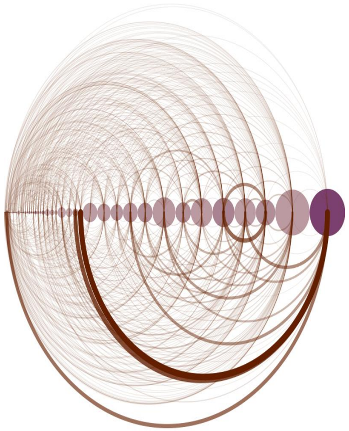
A map of 63,799 cross-references found in the Bible. The bottom bars represent number of verses in the given chapter. Color of arcs represents the distance between the two chapters.

http://www.chrisharrison.net/index.php/Visualizations/BibleViz

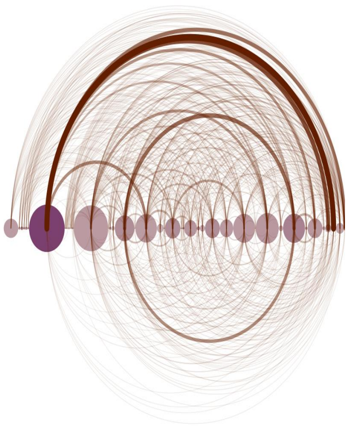
Arc Diagram



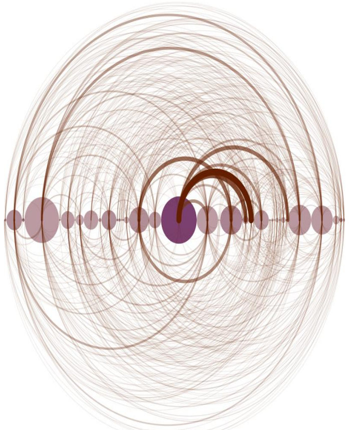
Sorted by the amount of incoming references



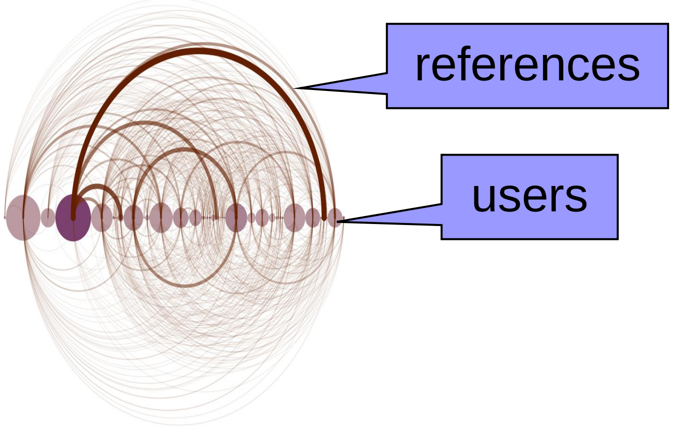
Sorted by the amount of outgoing references



Sorted by rate of incoming/outgoing references



Sorted by user name

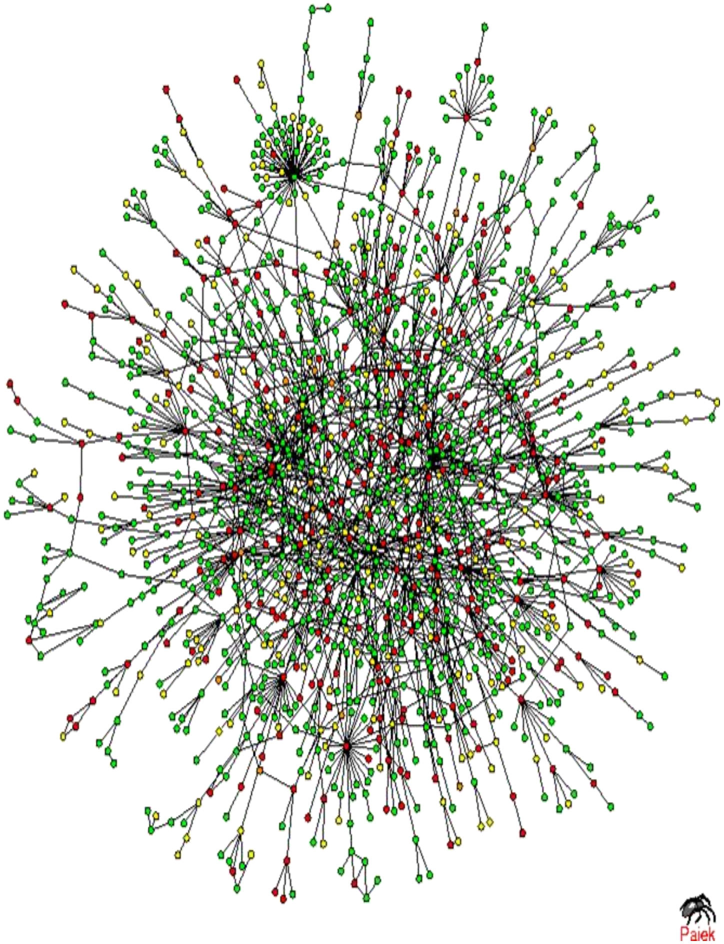


Unsorted

- Visualization of IRC communication behavior: Who is talking to whom?
- Arcs are directional and drawn clockwise:
  - In the upper half of a graph they point from left to right, in the bottom half from right to left
  - Arc strength corresponds to the number of references from the source to the target
- This visualization favors strong social connections over sociability: Frequent references between the same two users feature more prominently than combined references from several sources to a single target.
- http://datavis.dekstop.de/irc_arcs/

Centralized Burst

- Visualization with strong central tendency
- Can reveal highly connected objects (hubs) which usually correspond to objects with high importance
  - □ e.g., in a gene network, hubs are interesting points for targeting new drugs
    - Disabling a central gene probably will not allow the organism to adapt

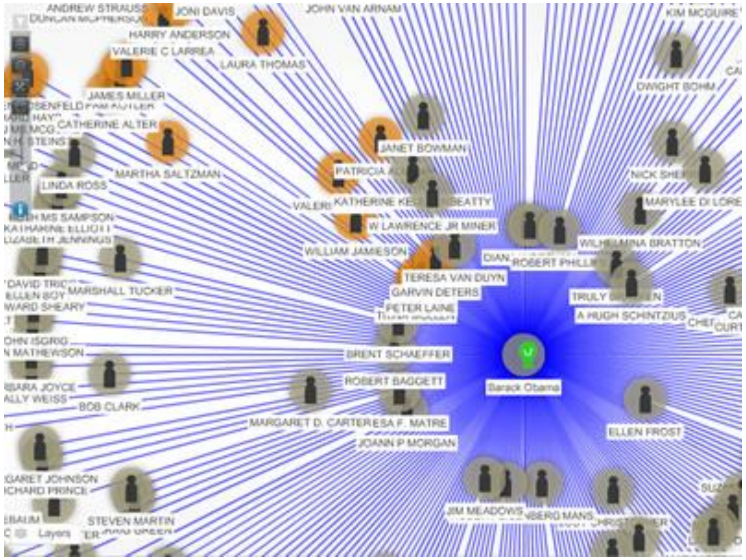


A map of **protein-to-protein interactions of a yeast**

source: H. Jeong. et al. “Lethality and Centrality in Protein Networks”, Nature, no. 411, 2011: 41-42

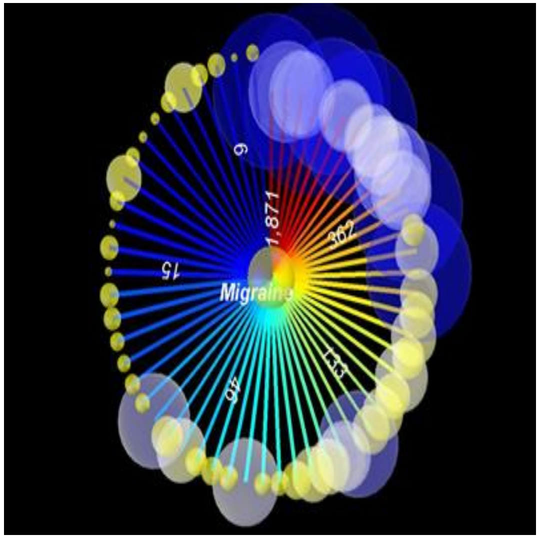
Centralized Ring

- Topology suitable for situations where we inspect a relation of multiple objects to one object
- Not very suitable for Big Data



A sociogram of individual donations in Asheville, North Carolina, to Barack Obama's 2008 presidential campaign.

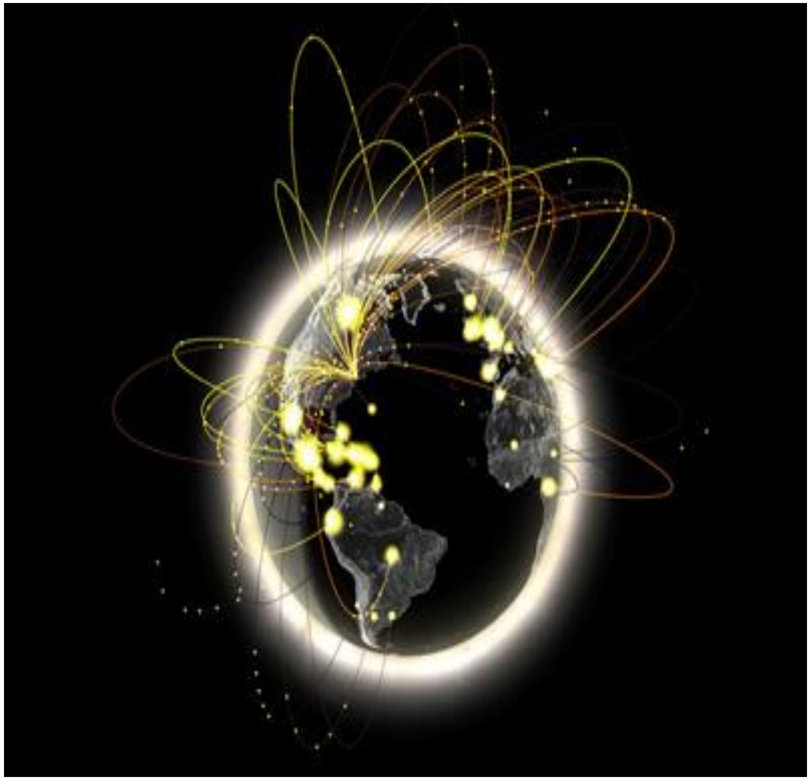
http://www.visualcomplexity.com/vc/project.cfm?id=613



A map of genetic overlap between migraine and about 60 other diseases. Each of the circles represents a different disease, and the size of the circle corresponds to the size of patient samples, ranging from 46 to 136,000, of those suffering from the disease.

Globe

- Globe visualizations are basically projections of other topologies on a globe

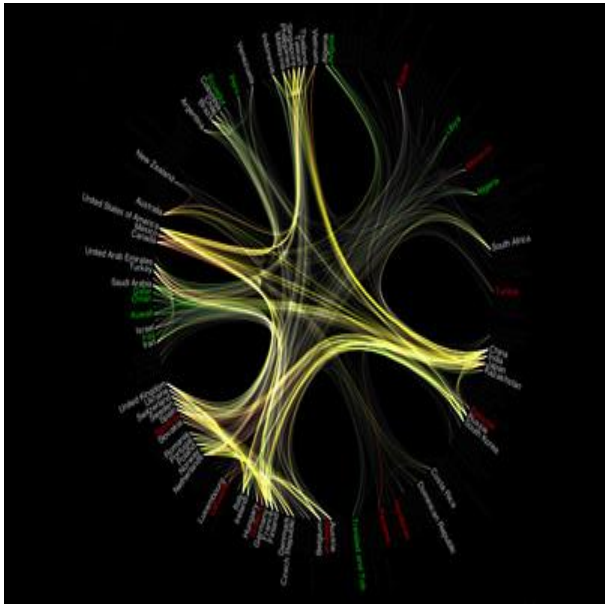


- The global exchange of information in real time by visualizing volumes of long distance telephone and IP data flowing between New York and cities around the world.
- How does the city of New York connect to other cities? With which cities does New York have the strongest ties and how do these relationships shift with time? How does the rest of the world reach into the neighborhoods of New York? The size of the glow on a particular city location corresponds to the amount of IP traffic flowing between that place and New York City. A greater glow implies a greater IP flow.

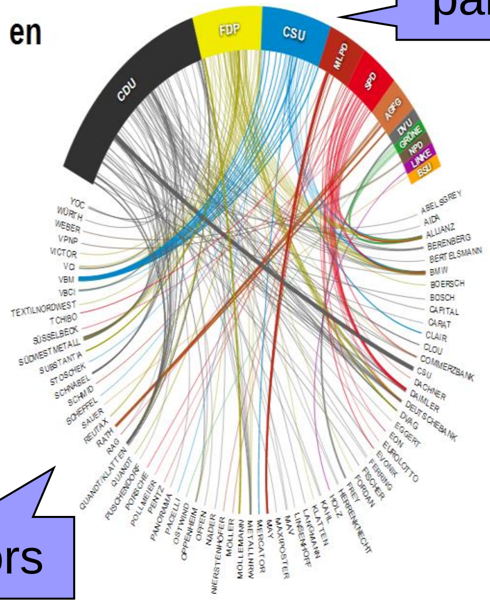
http://www.aaronkoblin.com/work/NYTE/index.html

Radial Convergence

- Also known as radial chart
- Actually a 360 arc diagram



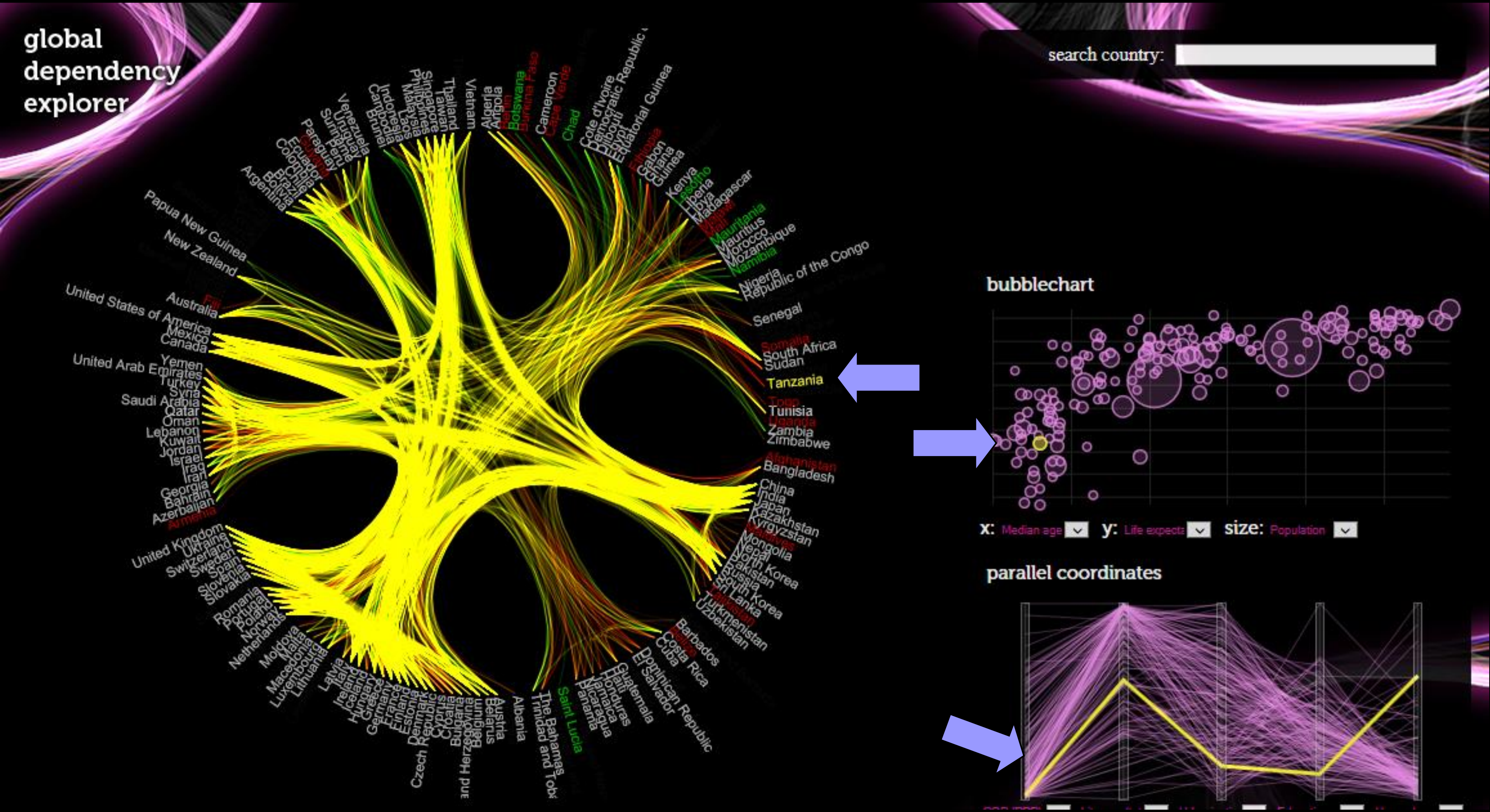
Tracking the commercial ties between most countries across the globe.
http://cephea.de/gde/



parties

donators

Money flow from private donators to parties in the German Bundestag (house of the parliament).
http://labs.vis4.net/parteispenden/



global dependency explorer

search country:

bubblechart

x: Median age y: Life expecta size: Population

parallel coordinates

Graph Drawing

- Spatial layout and graph drawing play a key aspect in information visualization
- Good layout needs to express the key features of a complex structure
- Graph drawing algorithms first agree on a criterion of what makes a good graph (and what should be avoided) and then run an algorithm driven by these criteria
- Generally, the primary goal is to optimize the arrangement of nodes so that strongly connected nodes appear close to each other
  - Most widely known graph drawing algorithms combine **force-directed** graph drawing and **spring-embedder** algorithms
  - The strength of a connection needs to be defined

Force-Directed Replacement

- Can be traced back to VLSI (very large scale integration) design = creating integrated circuits
  - Aim: optimize the layout of a circuit to a obtain as few number of crossings as possible
- Generally agreed on aesthetics criteria
  - **Symmetry**
  - **Even distribution of nodes**
  - **Uniform edge lengths**
  - **Minimization of edge crossings**
- Some of the criteria can be mutually exclusive
  - e.g., symmetric graph may require crossings which might be avoided

Force-Directed Replacement

- Replaces vertices in a graph by steel rings and edges by springs
  - Attractive force is applied to a pair of connected nodes
  - Repulsive force is applied to a pair of disconnected nodes
- Assigns forces among edges and nodes
  - Attractive: spring-like forces
    - Hook’s law = the force F needed to extend or compress a spring by distance X is proportional to that distance
      - $F = k . X$
      - k = characterizes stiffness of a string
  - Repulsive: forces of electrically charged particles
    - Coulomb's law: $F = k_e . (q_1 . q_2) / (r . r)$
      - q_1, q_2 = magnitude of charges of particles
      - k_e = Coulomb’s constant
      - r = distance of particles

Spring-Embedder Model (SEM)

- One of the most widely used models
  - Aim: uniform edge lengths and symmetry
- Observation: equilibrium state for the system of forces:
  - Edges tend to have uniform length (spring forces)
  - Nodes that are not connected by an edge tend to be drawn further apart (electrical repulsion)
- Algorithm:
  1. The model starts with a random initial step
  2. In each step, the vertices move accordingly to the spring forces to reduce the tension
  3. The optimal layout corresponds to the minimum-energy state of the system

SEM Optimization

Fruchterman and Reingold

- SEM can fail on very large graphs
  - Big Data
- Heuristics:
  - Attraction forces are computed only for neighboring nodes
    - Repulsive forces are computed for all pairs of nodes
      - A bit differently from SEM (optimization)
  - Optimization process is carried out iteratively
    - Controlled by temperature parameter
      - A similar way to **simulated annealing**
    - Only about 50 iterations are used

Simulated Annealing

higher level heuristics

- Probabilistic metaheuristics
- Problem of locating a good approximation to the global optimum of a given function in a large search space
    - Avoiding getting stuck in a local suboptimum
- Inspiration comes from annealing in metallurgy
    - Heating and controlled cooling of a material to increase the size of its crystals
- Idea:
    - The state of a system is randomly modified
        - Moves from state s_1 to s_2
        - s_2 does not need to be better than s_1
    - The probability of moving to a worse solution is given by temperature T which is gradually lowered

Graph Drawing Challenges

- Current algorithms are inefficient in incremental updating of the layout
  - i.e., one needs to redraw the whole layout when adding/removing a single node
- Networks with heterogeneous link types or node types cannot be efficiently handled
  - e.g., having users of a social network sharing various type of content (images, posts, video) and forming various types of relations (liking, tagging)
- Majority of algorithms focus on strong ties (heavyweight links)
  - Weak ties can be surprisingly valuable because they are more likely to be the source of novel information
    - e.g., hearing about a new job offering is an example of weak link with great social impact

Graph Drawing Challenges

- Scalability
  - Big Data related challenge
  - Problematic scaling as the size and density of the network increases
    - i.e., Big Data are also difficult to visualize
- Limited screen resolution
  - Big Data related challenge
  - Sometimes we simply do not have enough pixels to visualize a complex large-scale network
    - Zoomable interfaces, fish-eye views, …
    - In general solvable by interactivity

Graph Drawing Challenges

Scalability

- Solution with dense or large-scale networks can be partially solved by **reducing the complexity of the information** to be visualized
- Link reduction techniques
  - □ Pruning the original network
- Clustering
  - □ Dividing the network into smaller components and treat them individually
    - Inefficient if the graph contains large components
- Dimension reduction

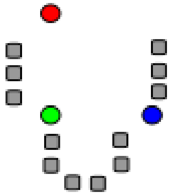
Link Reduction Techniques

- **Removing low weight links**
  - Imposing a link weight threshold ⇒ only link weights above the threshold are considered
  - Does not take into account the structure of the network
- **Minimum spanning tree**
  - Link reduction to $N - 1$ edges (on network of N nodes)
- **Network scaling algorithms**
  - e.g., pathfinder network scaling
    - Extracts paths of length at most Q
    - Exploits Minkowski metric
      - Generalization of Euclidean distance

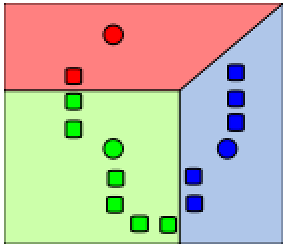
Clustering Techniques

- Goal: to divide a large data set into a number of sub-sets according to some given similarity measures
- Basic methodologies:
  - The choice is a trade-off between quality and speed
  - **Graph-theoretical**
    - Relies on a pre-computed distance matrix
    - Based on how objects are separated
    - e.g., single link, group average, complete link
  - **Single-pass**
    - Seed oriented clustering – clusters grow from a given number of data objects (seeds) chosen
  - **Iterative**
    - Iterative optimization of the clustering structure according to a heuristic function – k-means clustering
      - Repeating re-computation of centroids (step 3 + 4)
    - Unlike single-pass methods, the resulting number of clusters may not be known in advance
      - Hierarchical clustering

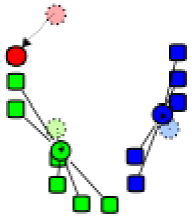
Step 1



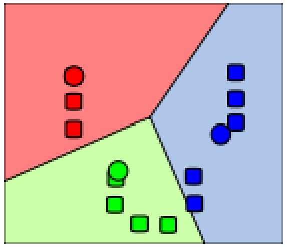
Step 2



Step 3



Step 4



Dimension Reduction (DR)

- When dealing with data which have relations expressed as a distance matrix only
  - Either we can use graph drawing methods
  - Or specialized dimension reduction techniques
- Idea: each **data point** consists of **multiple attributes** and the goal is to visualize **similar data points near to each other** in 2D space = **projection** from a multidimensional space **into a 2D space**
  - Generally difficult problems since in general distance space is not metric
    - Unlike Euclidian space

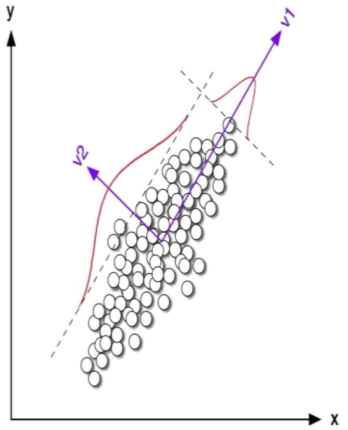
Dimension Reduction

Multi Dimensional Scaling (MDS)

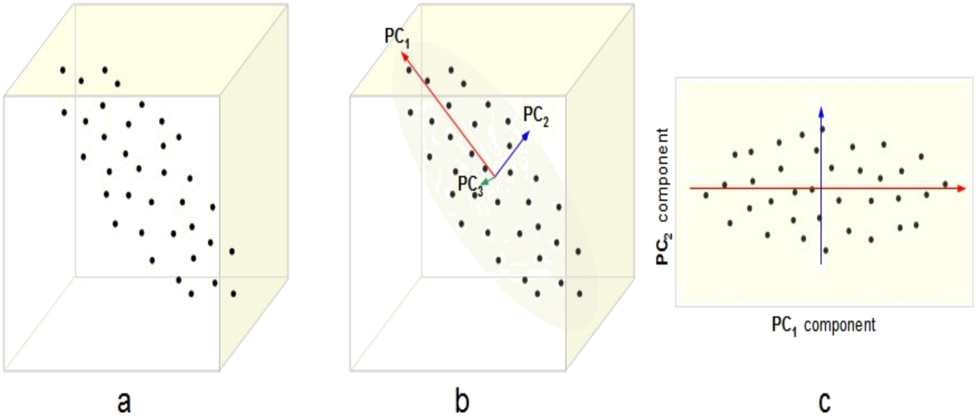
- The goal of MDS is to find a representation of data points (in nD) in the target space (in 2D) which resembles the mutual distances in the original space as close as possible
- Algorithm
  1. Generate an **initial layout**
  2. **Iteratively reposition** the data points so that the value of an error function for the current projection decreases
     - Error function = sum over all pairs of objects (distance in nD space – distance in 2D space)
  3. **Stop after given number** of iterations
- Very time demanding, especially for large datasets since all the distances need to be computed in every step
  - □ Optimizations have been introduced involving mainly parallel processing

Dimension Reduction

Principal Component Analysis (PCA)

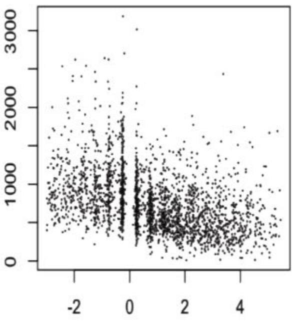


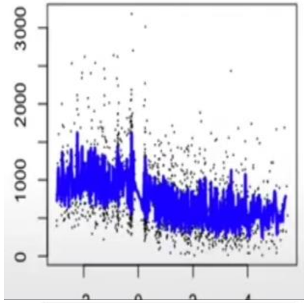
- Finding a linear transformation which tries to keep as much variability in the data as possible
- Identifies new basis vectors which maximize the amount of information kept after transformation onto the new basis
- New basis vectors correspond to the eigenvectors of the covariance matrix
  - The order of an eigenvalue/eigenvector specifies its informativeness ⇒ two first eigenvectors define a projection into 2D space keeping most of the information present in the data



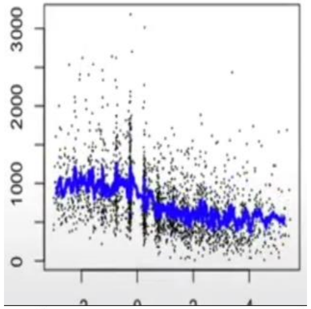
Smoothing

- Revealing patterns in large data
  - The patterns can be partially visible but not evident
- Techniques
  - **Moving average**
    - Representing trend using local averages
    - Sliding window and averaging values over the values
  - Locally weighted scatter plot smoothing (**LOWESS**)
    - Weights for the data points decline with their distance from center point according to a weight function
  - …

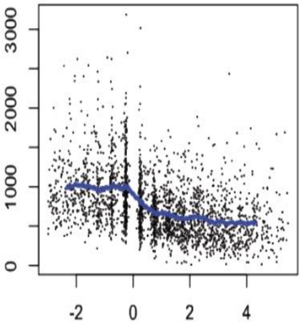




4 points window



20 points window



200 points window

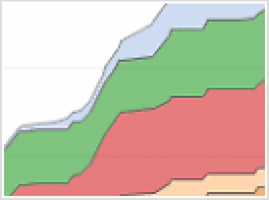
Data Visualization Tools

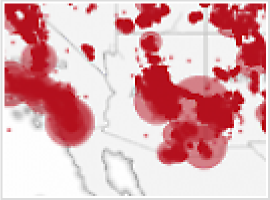
Expenses Report
sas
Unit Yield (rate), Unit Reliability by Geo
Revenue, Profit, Expenses by Date by Year
Correlation of selected measures
Frequency of Unit Age and Unit Reliability
Treemap of Margin, Sales by Channels

- Analytics and visualization tools
  - Standard statistical packages
    - R, Matlab
      - Customizability according to specific needs
  - Specialized data analytics/visualization solutions
    - SAS, IBM Cognos
      - Limited by the design
      - Ready-to-use solutions on top of a data warehouse
- Visualization tools
  - Tableau, Many Eyes (IBM), Circos, Visual.ly
- Trend: to bring the visualization and analysis to common users (not only data scientists)
  - Easy-to-use software
  - Web interfaces allowing instant sharing of visualizations
  - Drag and drop interfaces

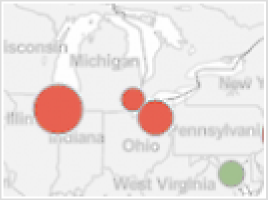
Data Visualization Tools

Tableau





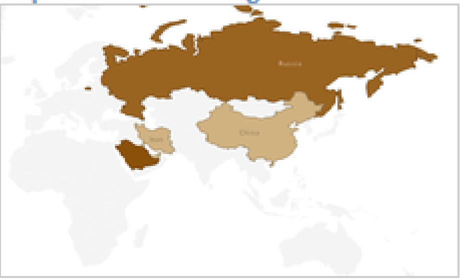
- http://www.tableausoftware.com/
- Data engine (analytics database) allowing ad-hoc analysis of massive data (millions of records)
- Products
  - □ Tableau Desktop – drag-and-drop tool, multiple views in a single dashboard, highlighting and filtering data, connection to live data
  - □ Tableau Server – sharing data and visualizations and work in progress
  - □ Tableau Online – hosted version of Tableau server allowing fast sharing of dashboards
  - □ Tableau Public – embedding of live visualizations into web pages
    - http://www.tableausoftware.com/public/gallery



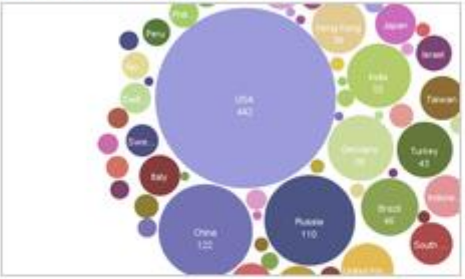
Data Visualization Tools

Many Eyes

- http://www-958.ibm.com/
- Data visualization experiment by IBM Research and the IBM Cognos software group
- Social networking application
  - □ Almost 175,000 visualizations as of today
    - http://www-958.ibm.com/software/analytics/manyeyes/visualizations
- Allows to update a dataset and create and share a visualization based on the data
  - □ Easy-to-use



Top 10 Oil Producing Nations

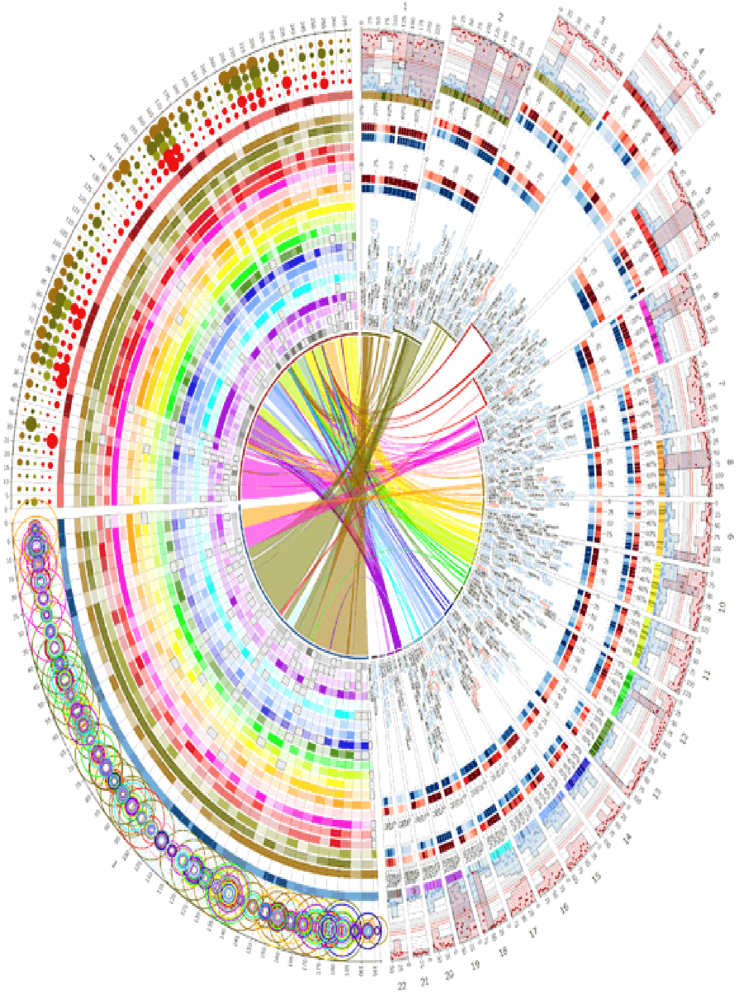


Billionaires in 2013

Data Visualization Tools

Circos

- http://circos.ca/
- Offline visualization tool specializing on circular visualizations
  - □ Exploring relationships between objects or positions
- User input: tab-delimited source file
- Output: circular visualization of the data in a form of an image
  - □ SVG, PNG



Programming Visualizations

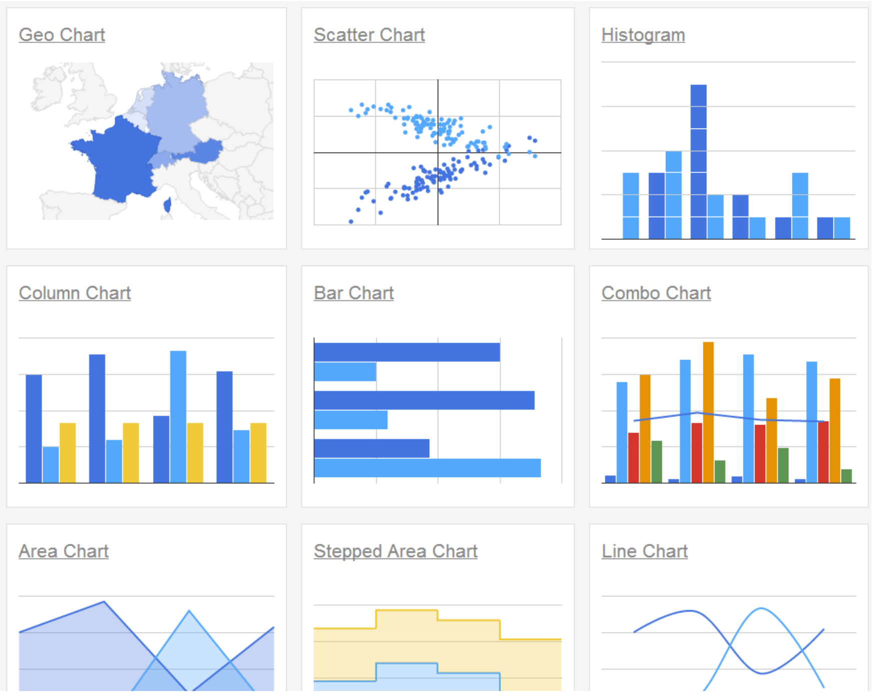
JavaScript Libraries

- D3 – Data-Drive Documents
  - □ Efficient manipulation of documents based on data
    - CSS3, HTML5, SVG
  - □ Comes packed with
    - Algorithms and layouts (force layouts, trees, …)
    - Drawing functionality (d3.svg)
    - jQuery-based selections
  - □ Hundreds of visualizations with source codes at http://bl.ocks.org/mbostock
- Raphael
  - □ More a graphics library
  - □ Every graphical object is a DOM object ⇒ JavaScript event handlers can be attached to it

Programming Visualizations

Google Charts API

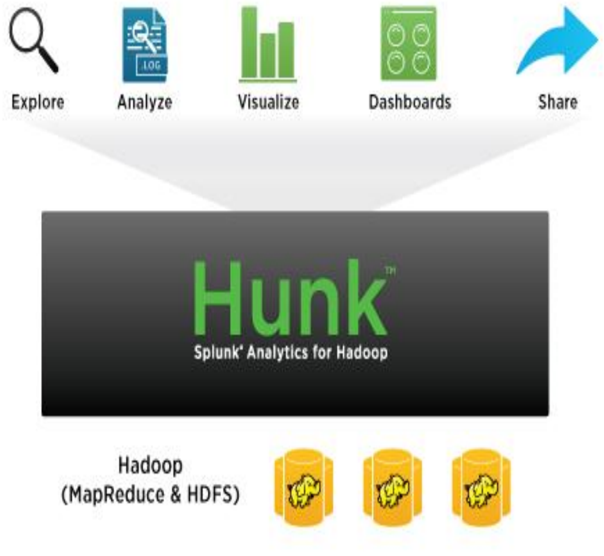
- Set of JavaScript classes
- Extremely simple to use:
  1. Load libraries
  2. List data to be charted
  3. Select chart type
  4. Create object with given id
  5. Create div element with the id

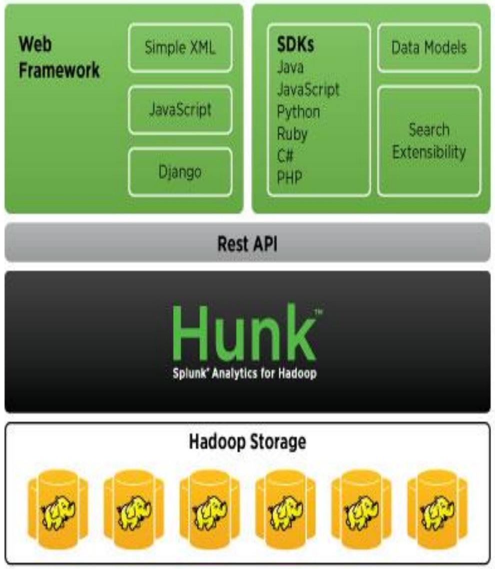


https://developers.google.com/chart/

Hunk

- http://www.splunk.com/hunk
- Analytics tools are not designed for the diversity and size of Big Data sets
- Hunk = Hadoop more accessible to common users
- Allows users to analyze and visualize historical data in Hadoop
  - Enables to detect patterns and find anomalies





Project Neo

- Dashboard atop datasets
  - Allowing untrained workers to create visualizations and discover patterns and insights from raw data
- IBM’s effort to make data visualization available not only to data analytics or scientists
- To be launched in 2014
- Focus on speed
  - Built on top of Rapidly Adaptive Visualization Engine

Data Visualization Techniques

Course Topics

Lectures:

- Map and geographical data analysis and visualization
- Time series analysis and visualization
- Network data analysis and visualization
- Interactive inforgraphics
- 3D visualization
- Searching in visualizations
- ...

Labs:

- Creating visualizations
  - Probably using
    - Tableau
    - Circos
    - Visual.ly
- Programming visualizations
  - D3.js

References

- Edward R. Tufte: The Visual Display of Quantitative Information
- Edward R. Tufte: Envisioning Information
- Chaomei Chen: Information Visualization: Beyond the Horizon
- Manuel Lima: Visual Complexity: Mapping Patterns of Information