

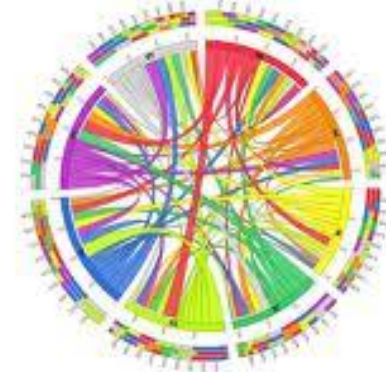
```

<?xml version="1.0" encoding="utf-8" ?>
<INFORMATIONS>
  <INFORMATION>
    <HEADER>
      <Name>Santosh.R</Name>
      <Domain>@AX.LOCAL</Domain>
    </HEADER>
    <ROWS>
      <ROW>
        <DETAILS>
          <ACCOUNT>108600</ACCOUNT>
          <COSTCENTER>112400</COSTCENTER>
          <DESCRIPTION>Row 1</DESCRIPTION>
        </DETAILS>
      </ROW>
    </ROWS>
  </INFORMATION>
  <INFORMATION>
    <HEADER>
      <Name>Anitha.E</Name>
      <Domain>@AX.LOCAL</Domain>
    </HEADER>
    <ROWS>
      <ROW>
        <DETAILS>
          <ACCOUNT>126556</ACCOUNT>
          <COSTCENTER>224212</COSTCENTER>
          <DESCRIPTION>Row 2</DESCRIPTION>
        </DETAILS>
      </ROW>
    </ROWS>
  </INFORMATION>
</INFORMATIONS>

```

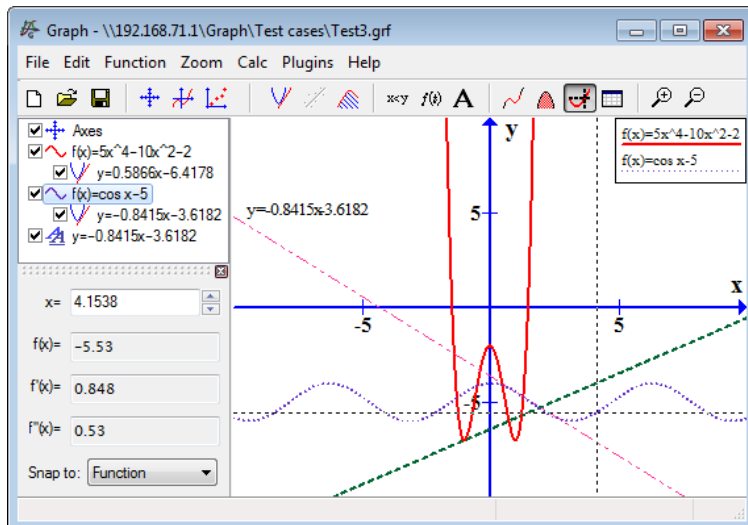
axblog4u.wordpress.com

	A	B	C	D	E	F	G	H
A	54	113	157	94	88	141	167	133
B	49	113	111	113	202	53	7	92
C	66	110	69	162	123	63	106	117
D	60	118	89	85	98	98	122	87
E	51	88	15	92	92	10	69	127
F	118	32	62	119	135	95	60	64
G	114	108	73	44	103	119	37	145
H	74	110	84	120	9	41	45	131

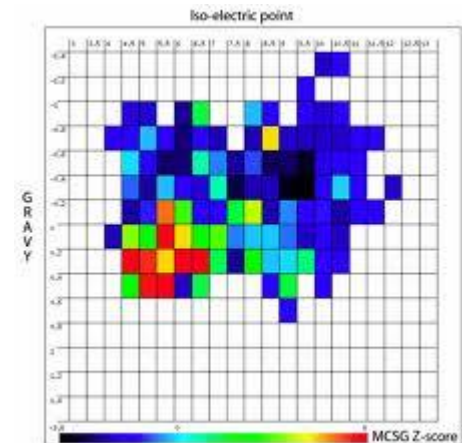


mkweb.bcgsc.ca

2. Input data characteristics



www.padowan.dk



technology.sbk.org

Process of data visualization

- Generating data
 - Measuring, simulation, modeling
 - Can be lengthy (measuring, simulation) and costly (simulation, modeling)
- Visualization (the rest of the visualization pipeline)
 - Visual mapping, rendering
 - Can be fast or slow, depending on hardware and implementation
- Interaction (user feedback)
 - How the user can interact with the visualization

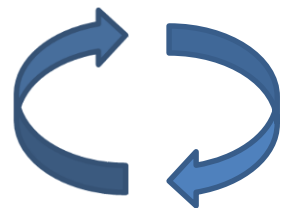
Passive visualization

- The following three steps are strictly separated
 - Generating data – after finishing this phase
 - Off-line visualization
 - Displaying the generated data
 - Result is a video or animation
 - Passive visualization
 - Exploration of the results of the previous phase



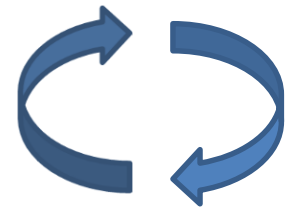
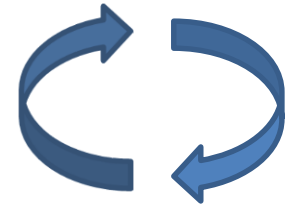
Interactive visualization

- Here only the generating data phase is separated
 - Off-line data generation
 - Interactive visualization
 - Generated data is available for interactive visualization
 - Options: selection, parametrization of visualization
 - Currently very popular technique

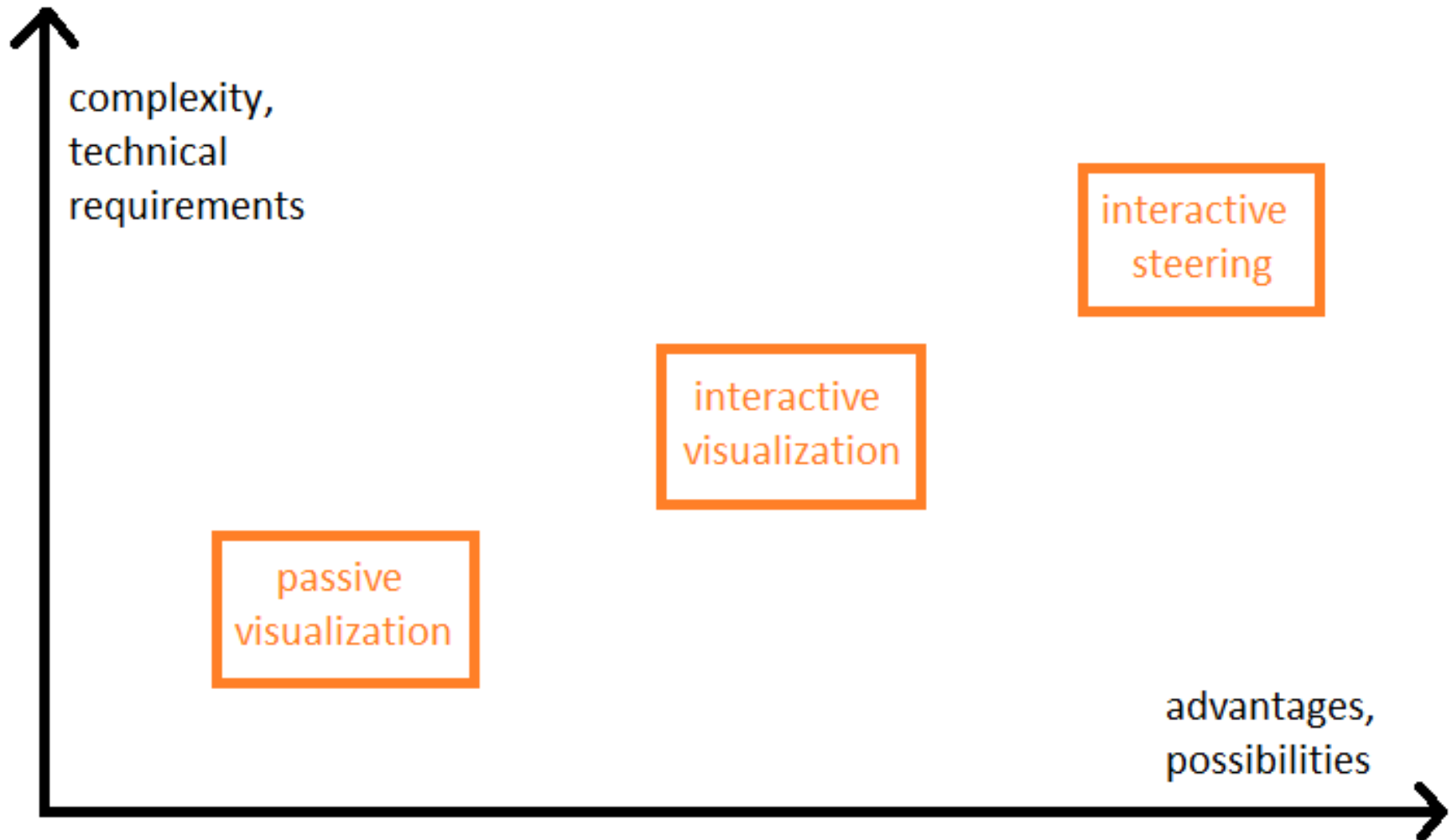


Interactive steering

- All three steps are connected
 - Generating data on the fly
 - Interactive visualization enabling real-time view onto data
 - Extended interaction
 - The user can control the simulation process, change design when modeling, etc.
 - Very complicated and costly process

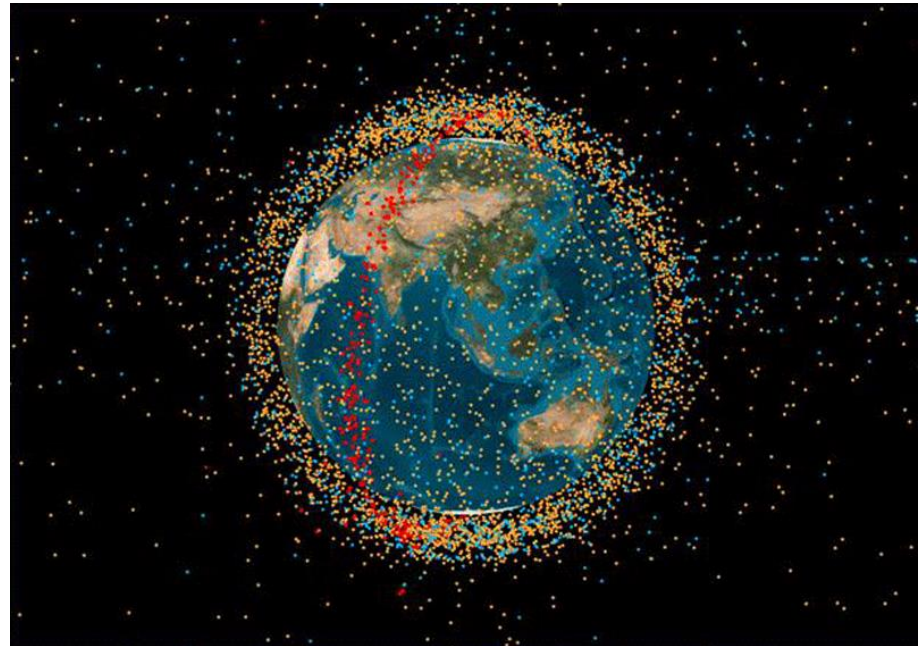


Comparison



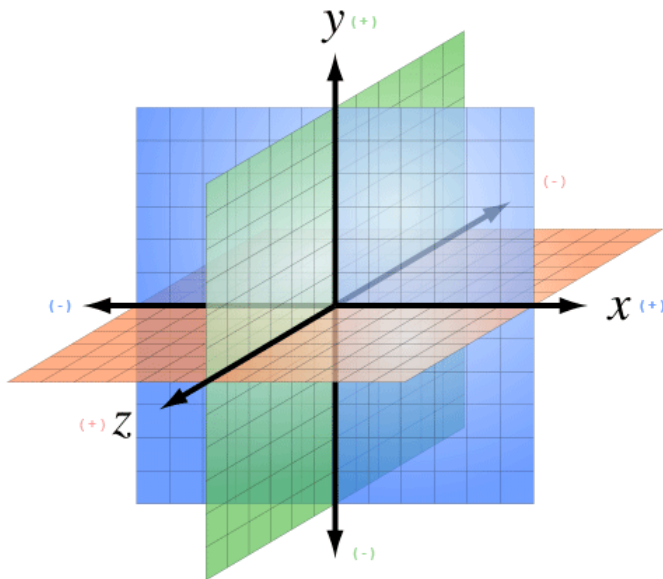
Data

- Central topic of visualization
- Data influences the selection of appropriate visualization technique (along with the user)
- Important questions:
 - Where data „lives“
(what is the **data space**)
 - **Type** of data
 - Which **representation** is meaningful



Data space

- Different properties
 - Dimensionality of data space
 - Coordinate system
 - Region of influence (local or global impact)



Data definition

- *Raw data x preprocessed data*
- Data item (r_1, r_2, \dots, r_n)
- Each r_i record contains m variables (v_1, v_2, \dots, v_m)
- v_i is often denoted as observation

Definition of variables

- *Independent variable iv_i*
 - not influenced by any other variable (e.g., time)
- *Dependent variable dv_j*
 - is influenced by one or more independent variables (e.g., temperature)

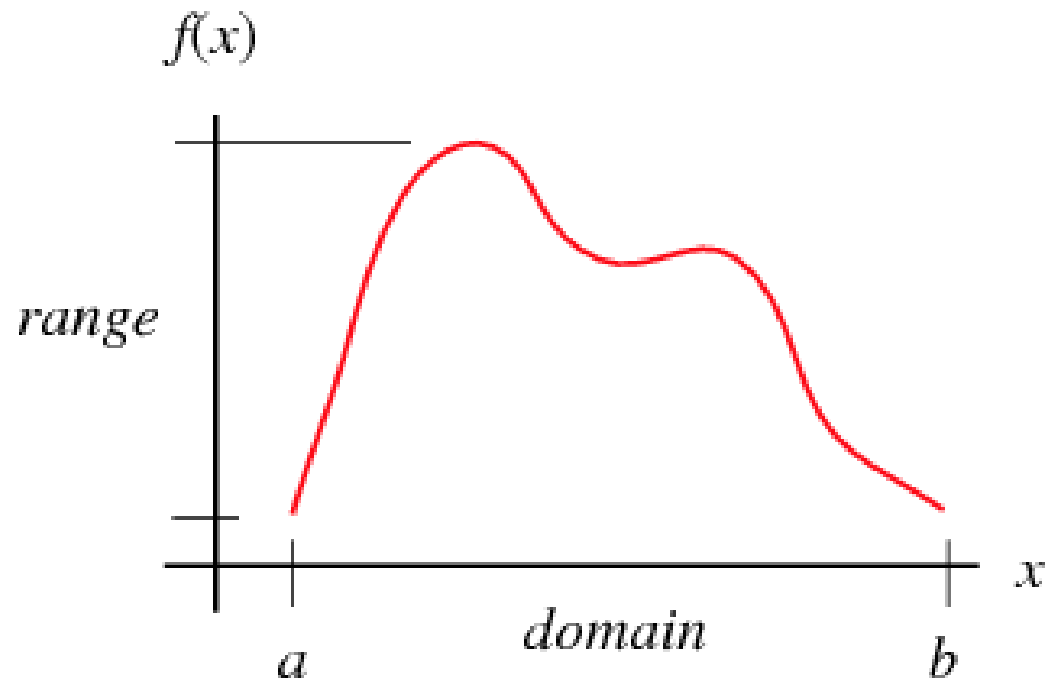
- Record can be represented as

$$r_i = (iv_1, iv_2, \dots, iv_{m_i}, dv_1, dv_2, \dots, dv_{m_d})$$

where $m = m_i + m_d$

Data generated by function

- Independent variables = domain
- Dependent variables = range



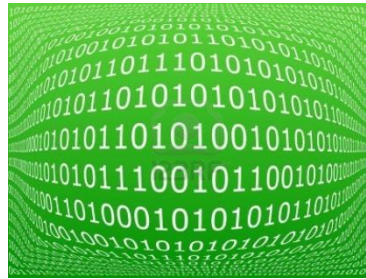
Types of variables

- Physical types
 - Characterized by the input format
 - Characterized by the type of possible operations
 - Example: bool, string, int, float,...
- Abstract types
 - Data description
 - Characterized by methods/attributes
 - Can be hierarchical
 - Example: plants, animals, ...

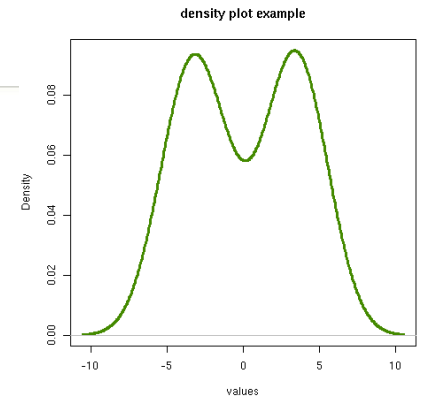
Data types

- **Ordinal**

- Binary
- Discrete
- Continuous



www.123rf.com

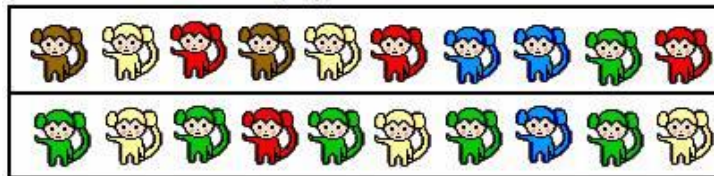


orgmode.org

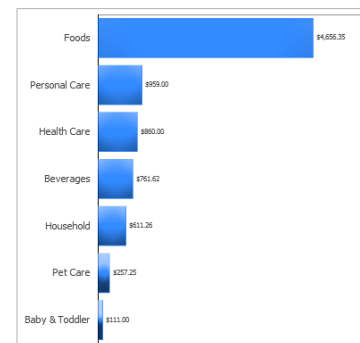
- **Nominal**

- Categorical
- Sorted
- Random

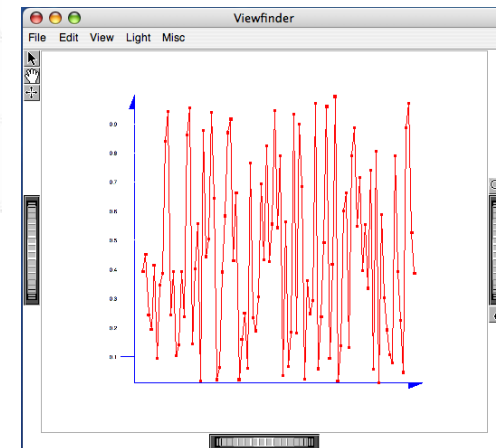
Different colored toys placed in two different shelves



www.icoachmath.com



tennysusantobi.blogspot.com



www.cincomsmalltalk.com

Scale

- 3 basic attributes:
 - Ordering relation on data
 - Distance metric
 - Existence of absolute zero
 - Fixing the minimal value of variable

ABSOLUTE ZERO



The only thing cooler is
not being a scientist.

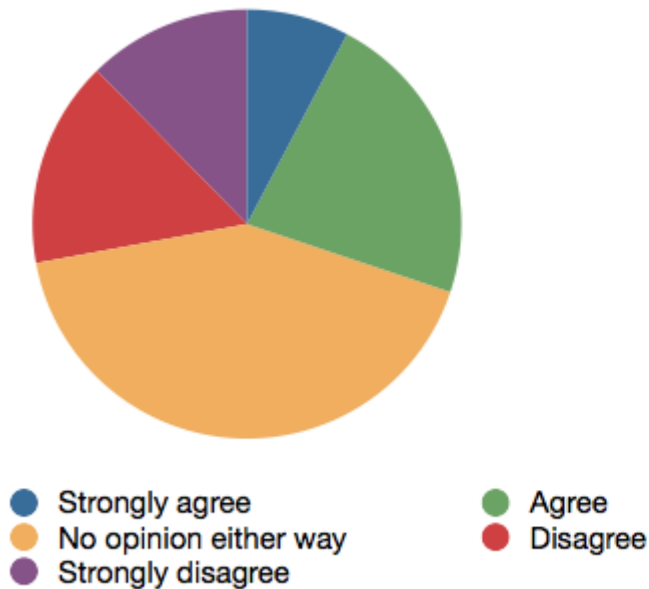
Data representation

- Depends on:
 - The presence of spatial domain
 - If it is not inherently in data, which domain to choose?
 - How the dimensions are used?
 - Data characteristics
 - Available visualization space (2D/3D)
 - Which part of data is in focus?
 - In which parts we can use more abstracted representation?

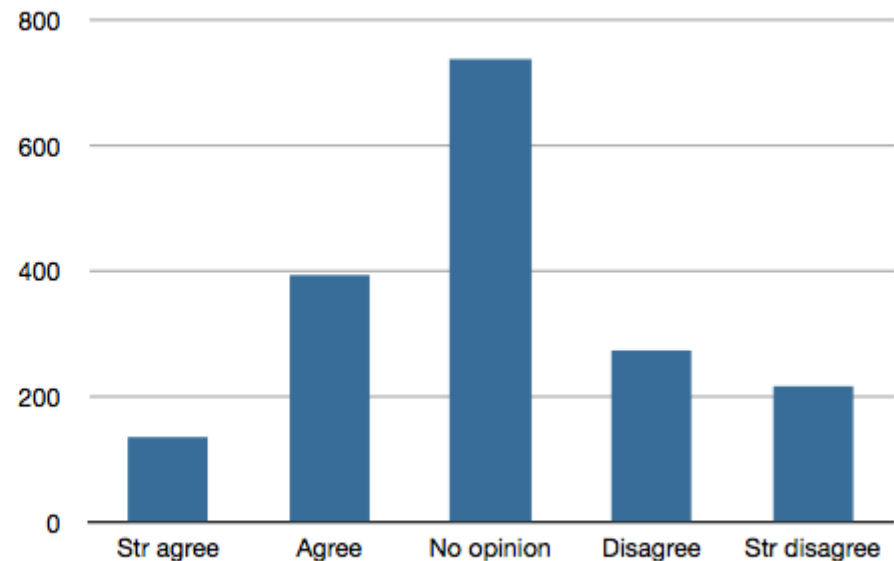
Examples

- Discrete data – set of values, visualization using bar charts, pie charts, ...

Level of agreement with the Tea Party: a pie chart

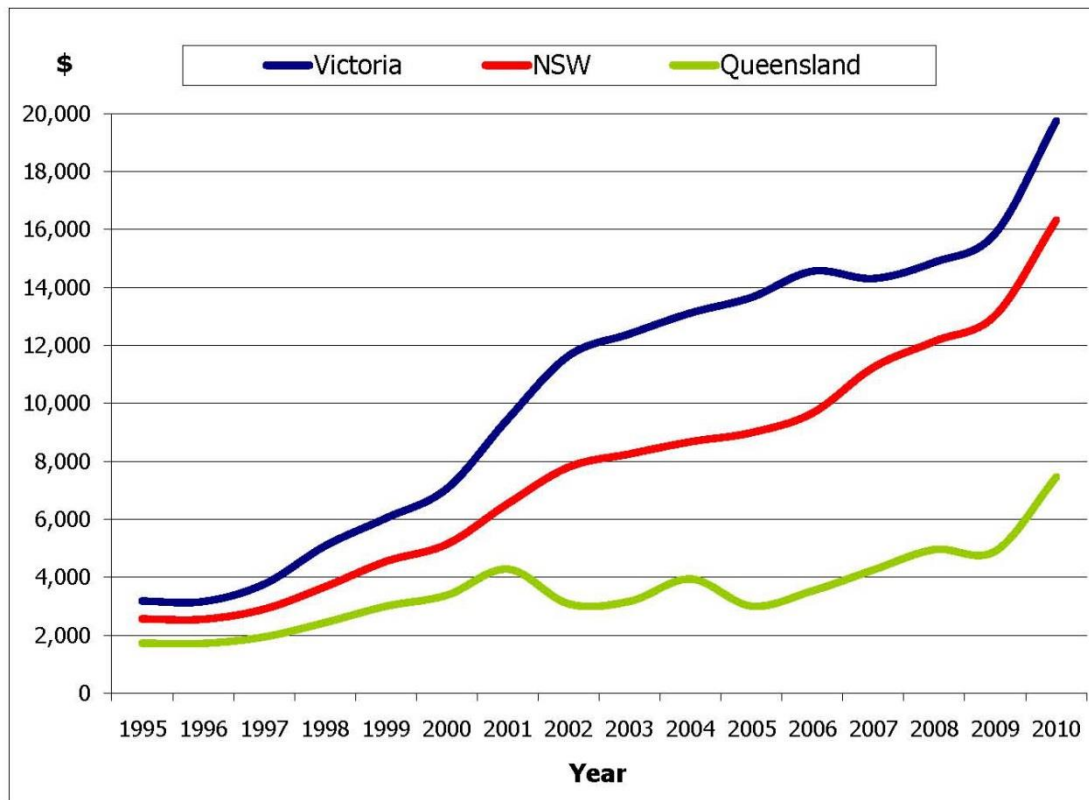


Level of Agreement with the Tea Party: a bar chart



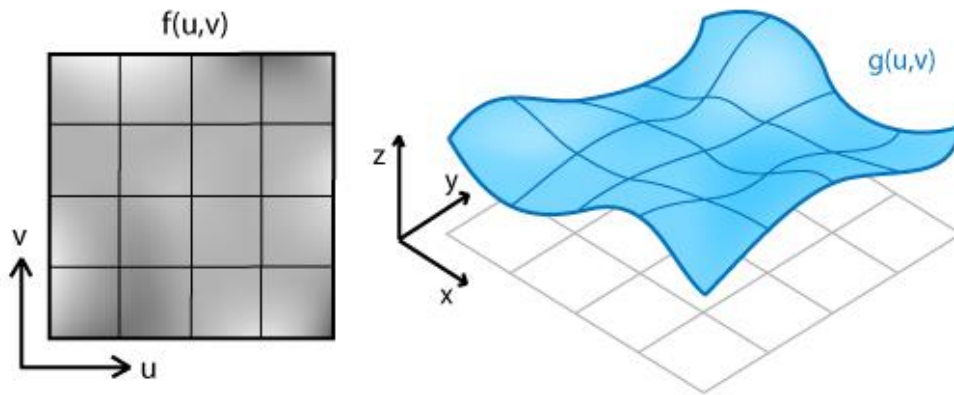
Examples

- Continuous data – function, visualization using graphs



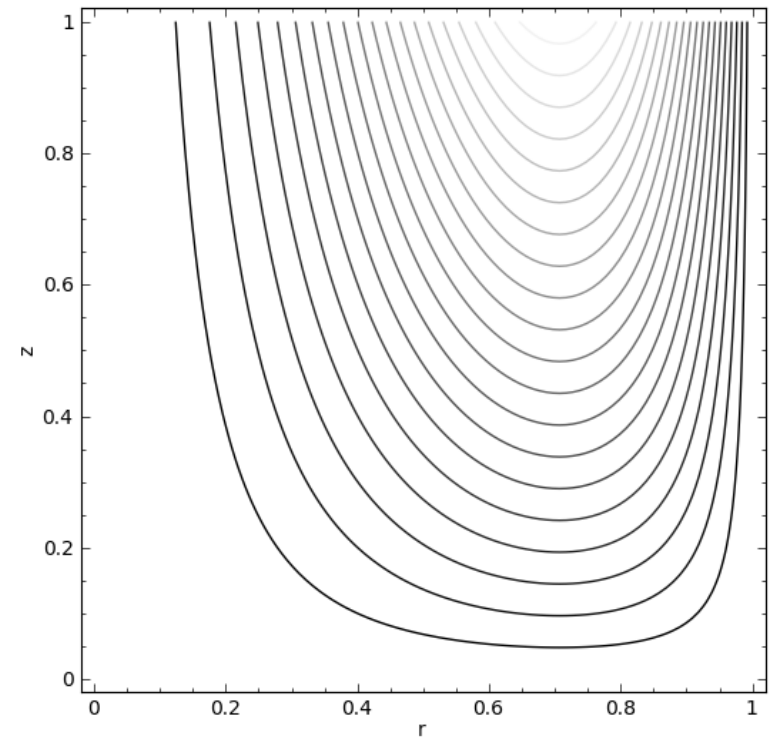
Examples

- 2D real numbers
 - Function of two variables, visualization using 2D height maps, contours in 2D, ...



$$g(u, v) \rightarrow (x, y, z) : \begin{cases} x = u \\ y = v \\ z = f(u, v) \end{cases}$$

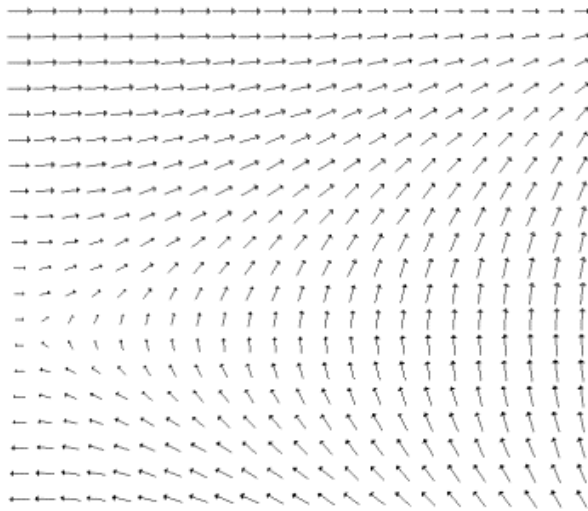
acko.net



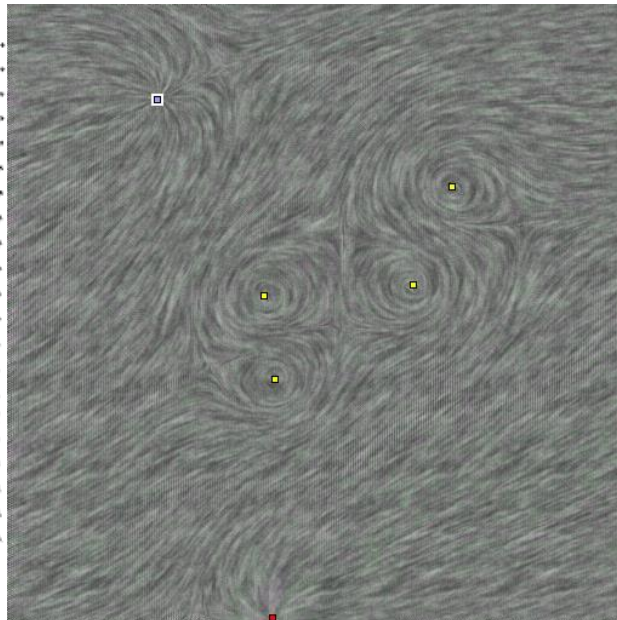
timothyandrewbarber.blogspot.com

Examples

- 2D vector fields, visualization using hedgehog plots, LIC (line integral convolution), streamlets, ...



csis.pace.edu



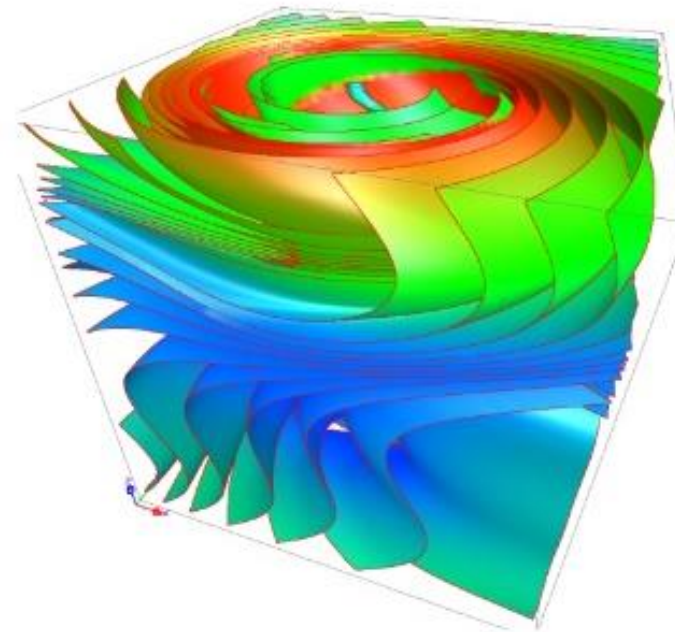
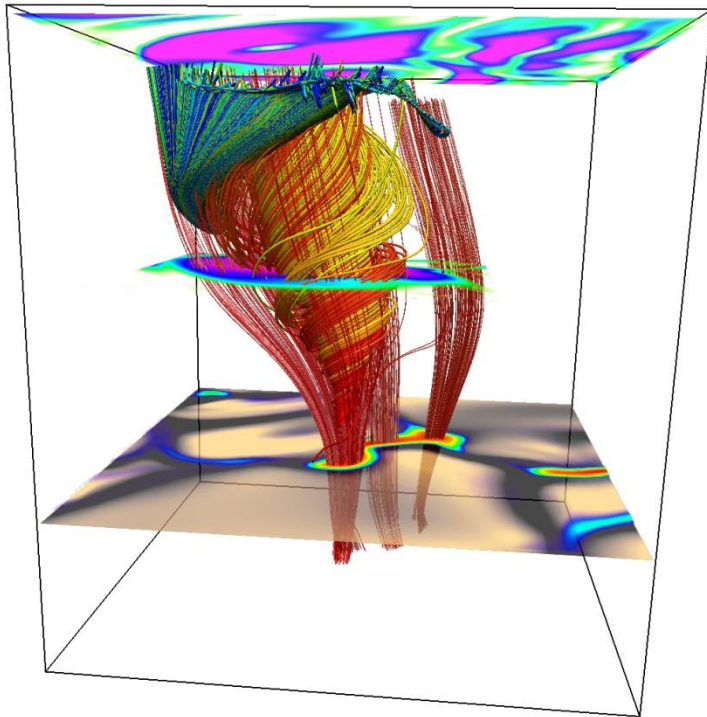
en.wikipedia.org



www.cg.tuwien.ac.at

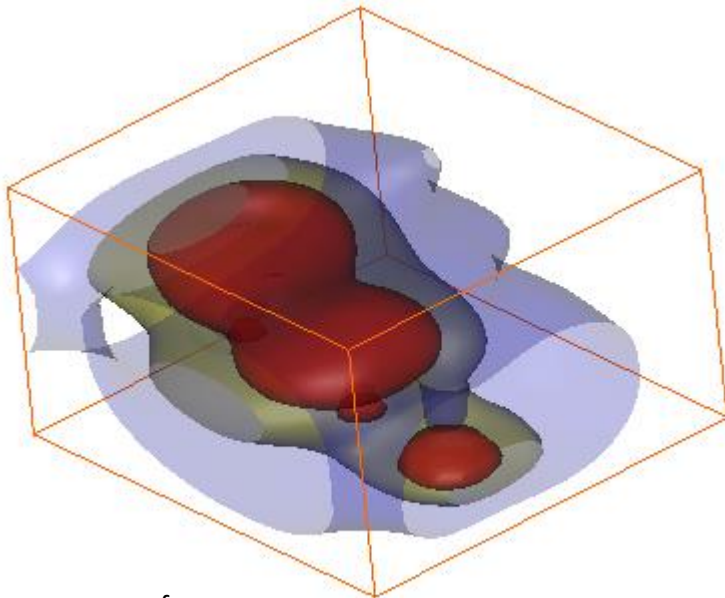
Examples

- Spatial data + time
 - 3D flow, visualization using streamlines, streamsurfaces

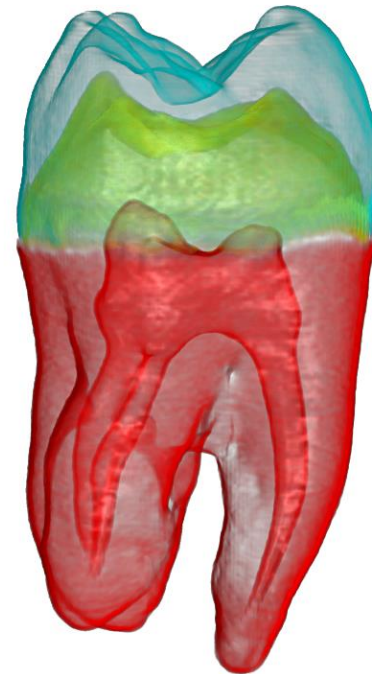


Examples

- Spatial data
 - 3D density, visualization using isosurfaces, volume rendering



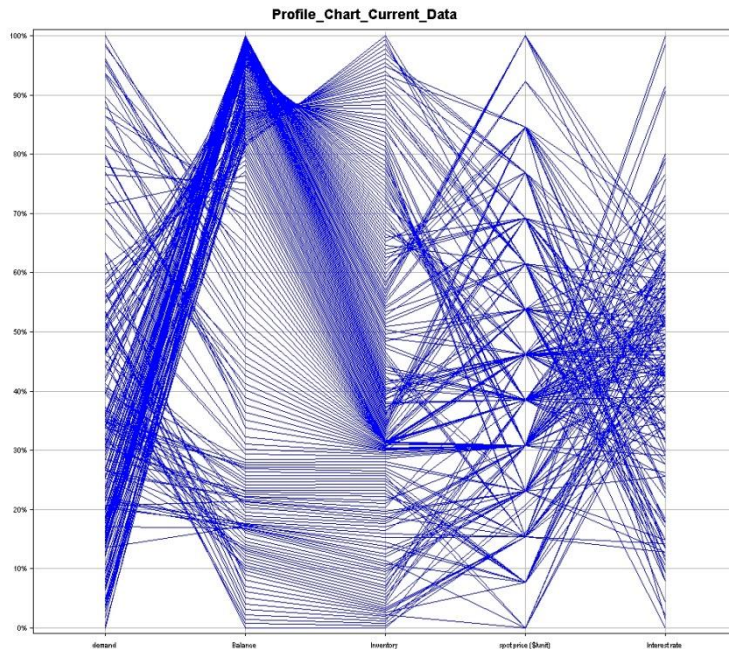
www.ssg-surfer.com



viscg.uni-muenster.de

Examples

- Multidimensional data
 - Set of n dimensions, visualization using parallel coordinates, glyphs, icons, ...



www.cs.umd.edu

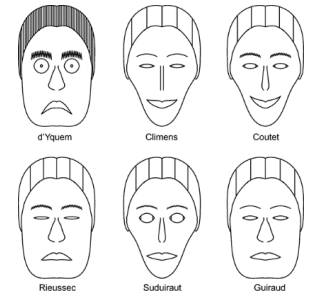
Maximum Values Face



Minimum Values Face



www.emeraldinsight.com



datamining.typepad.com

Structure inside and between records

- Data sets consist of:
 - **Syntax** – data representation (so called data model)
 - **Semantics** – relationships within one record or between records (so called conceptual model)
- Types of structures:
 - Scalars, vectors, tensors
 - Geometry and grids
 - Other forms

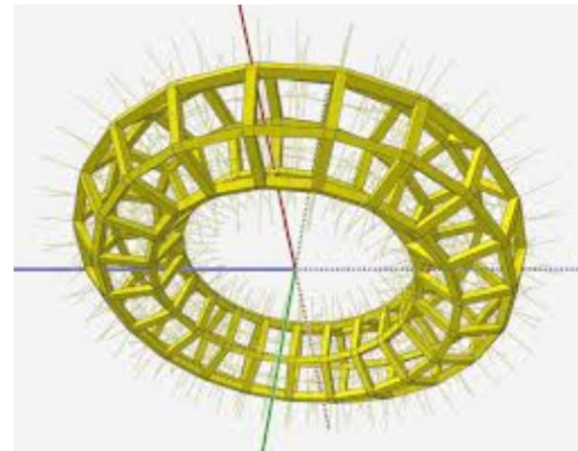
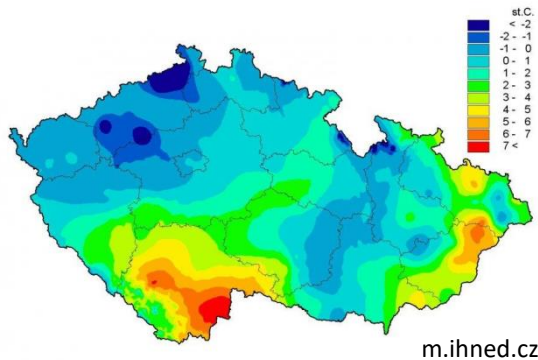


Scalars, vectors, and tensors

- **Scalar** = individual number in record
 - e.g., age
- **Vector** = composition of several variables to one record
 - e.g., point in 2D space, RGB
- **Tensor** = defined by its order and space dimension. Represented by field or matrix.
 - e.g., transformation matrix in 3D

Geometry and grids

- Geometry is represented using coordinates of records

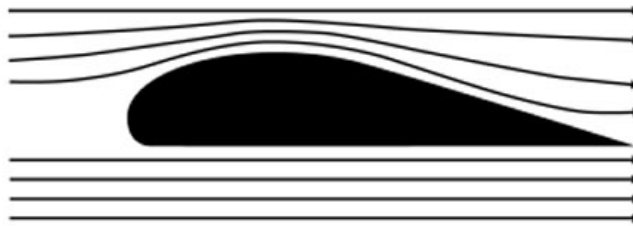


danielwalsh.tumblr.com

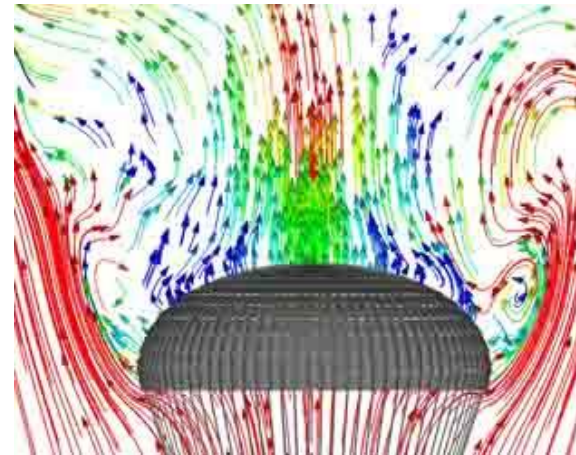
- Grid – geometry can be derived from the starting position, orientation, and step size in horizontal and vertical direction

Non-uniform geometry

- We need to store coordinates of all records – they cannot be derived

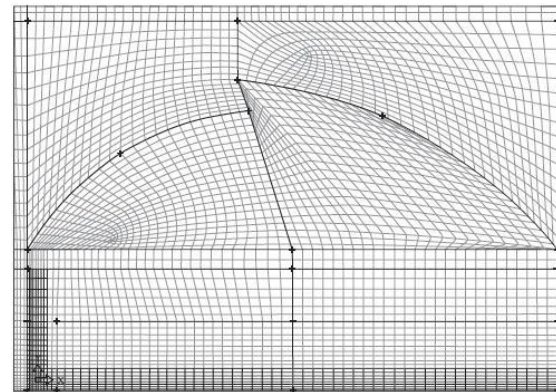


blog.nasm.si.edu



www.tafsm.org

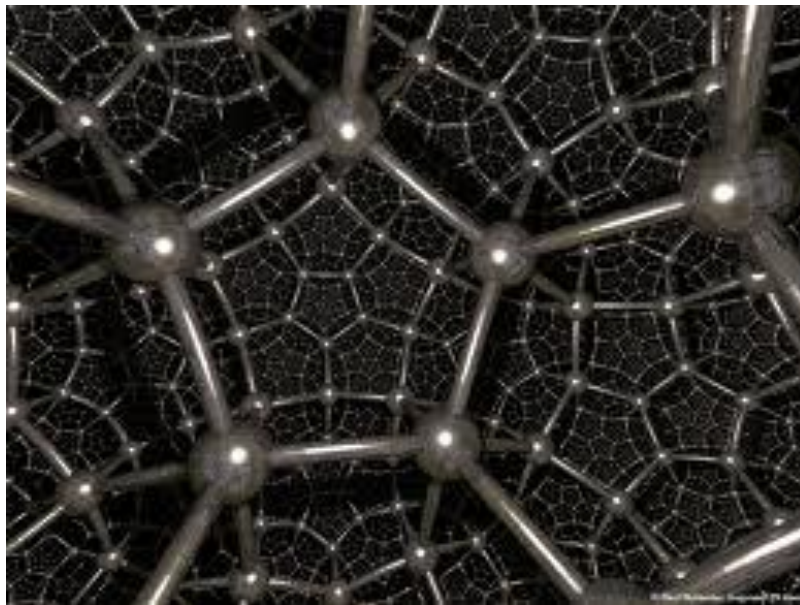
- Non-uniform grid



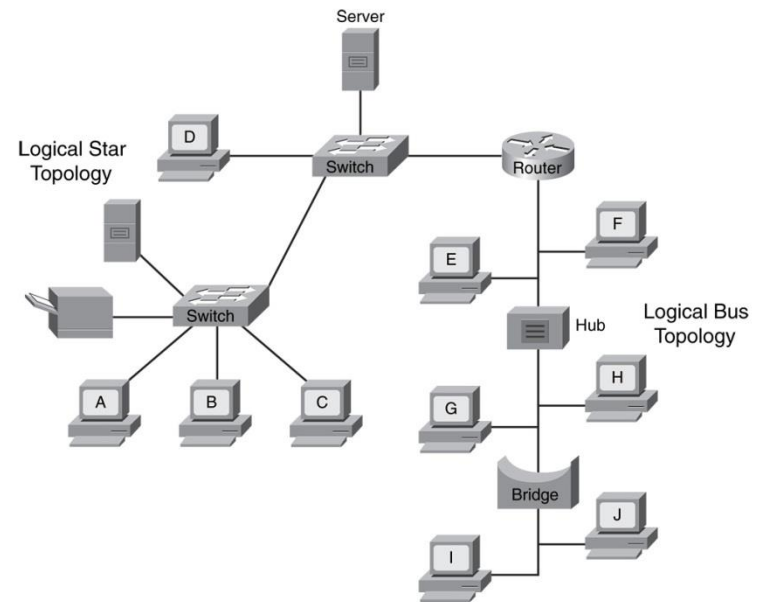
www.scielo.org.mx

Other types of structures

- Another important structure is **topology**
- It defines so called **connectivity**



www.bugman123.com



xpertnetworking.wordpress.com

- Important in resampling and interpolation

Time

- Enormous range of values (picosecs vs. millenia)
- Expressed absolutely or relatively
- Data sets containing time can have regular distribution (regular sampling) or irregular one (e.g., transaction processing – according to the time stamp of its execution)



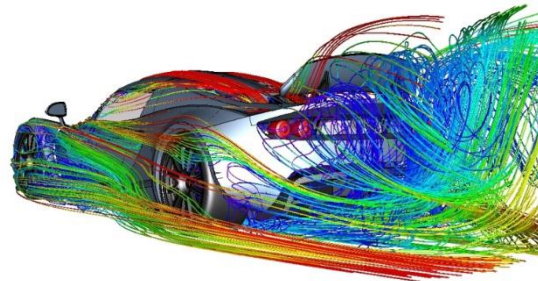
megworden.com

Other examples of structured data

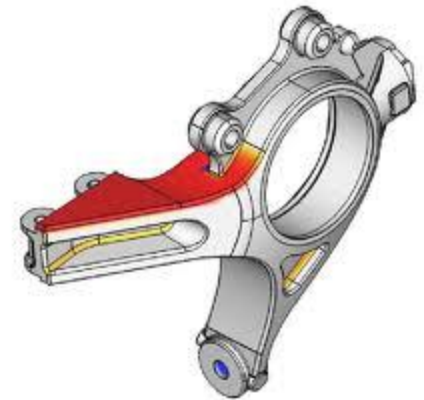
- Magnetic Resonance Imaging (MRI)
- Computational fluid dynamics (CFD)
- Financing
- CAD systems
- Counting people
- Social networks



www.impactlab.net



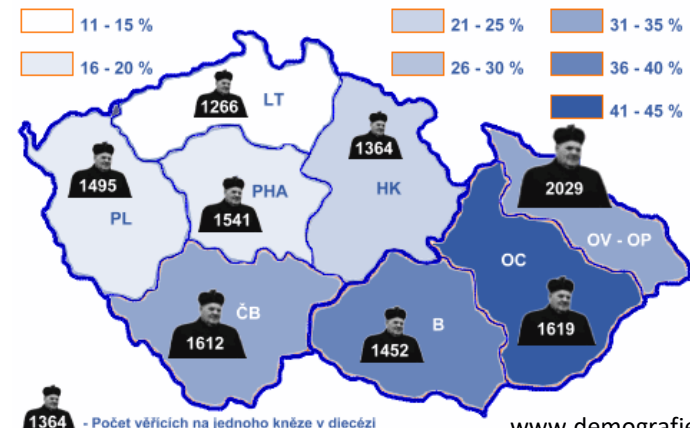
lotusenthusiast.net



www.mjmdesigns.co.uk



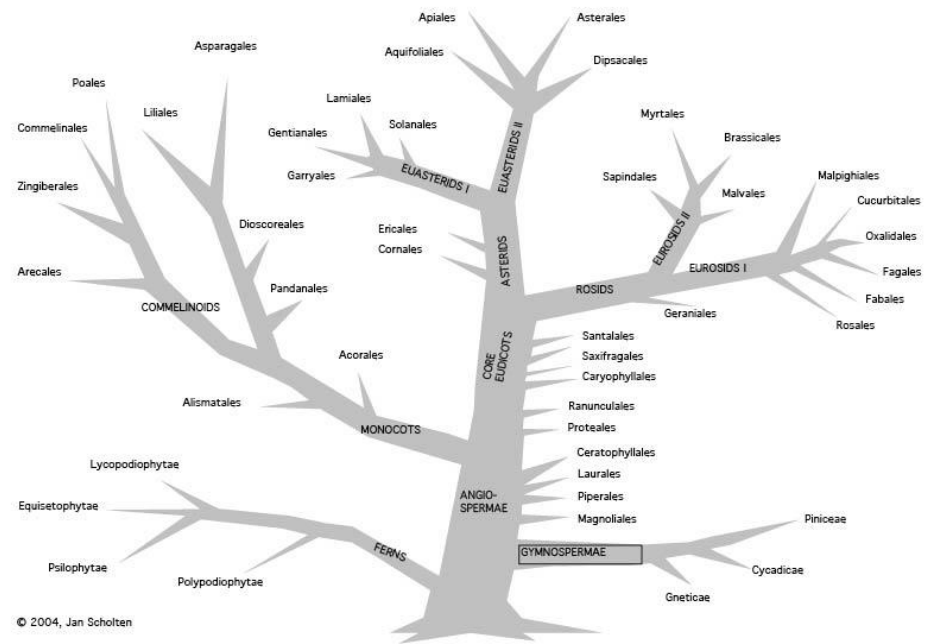
successfulworkplace.com



www.demografie.info

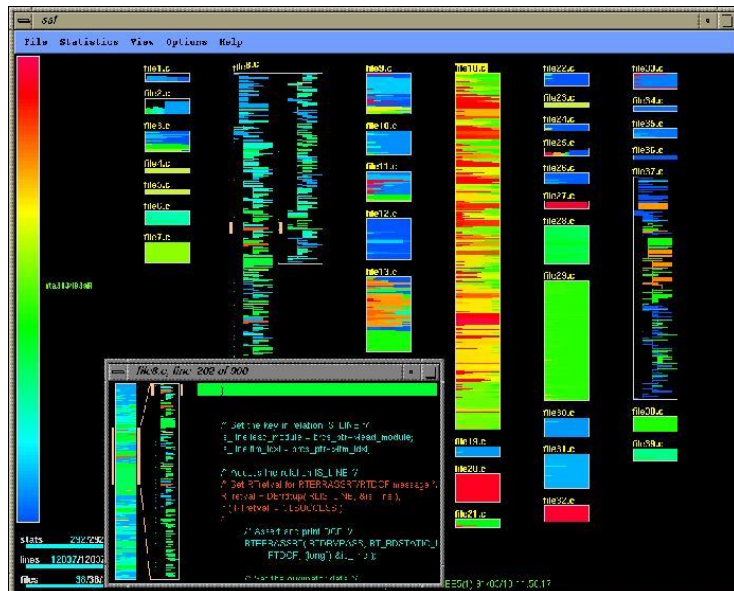
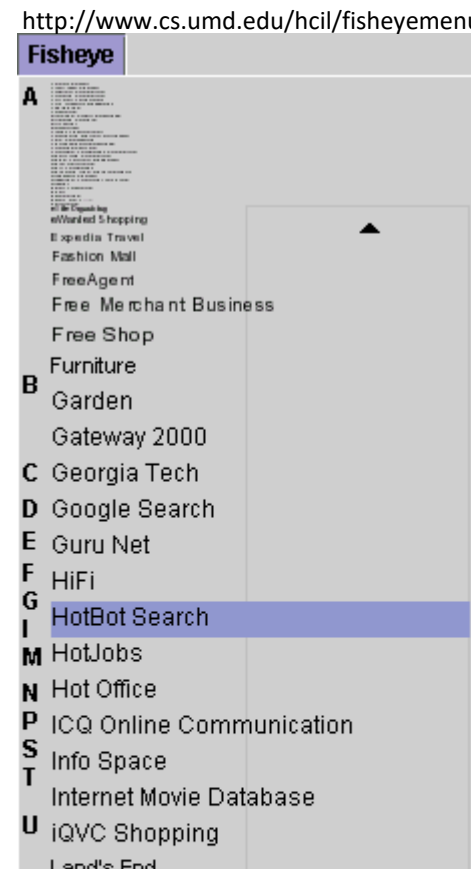
Taxonomy – 7 types of data

- 1D (linear sets and sequences)
- 2D (maps)
- 3D (objects, shapes)
- nD (relations)
- Trees (hierarchy)
- Networks (graphs)
- Temporal data



Linear data

- Long lists of items
 - Menu items
 - Source code
- Fisheye displays



<http://ds.cc.yamaguchi-u.ac.jp/~ichikay/pfp7/iv/pics/SeeSoft-line.jpg>

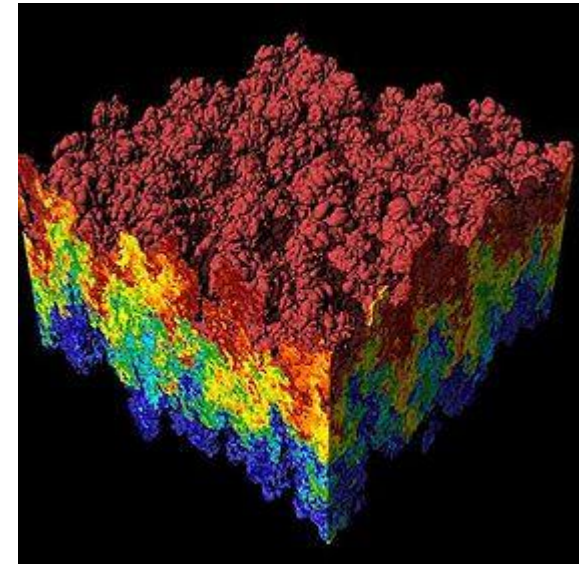
2D data - maps

- GIS (Geographical Information Systems)
 - Maps (e.g., Google Earth)
 - Spatial queries
 - Spatial data analysis

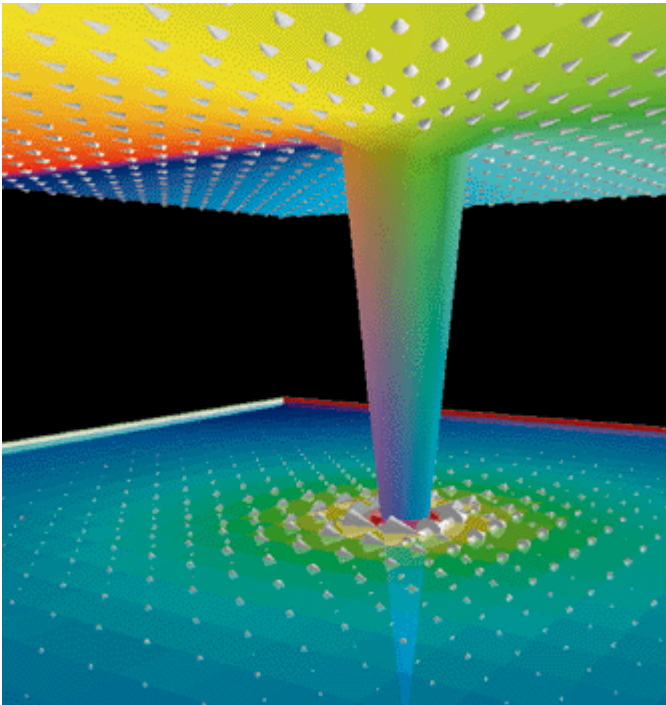


3D data

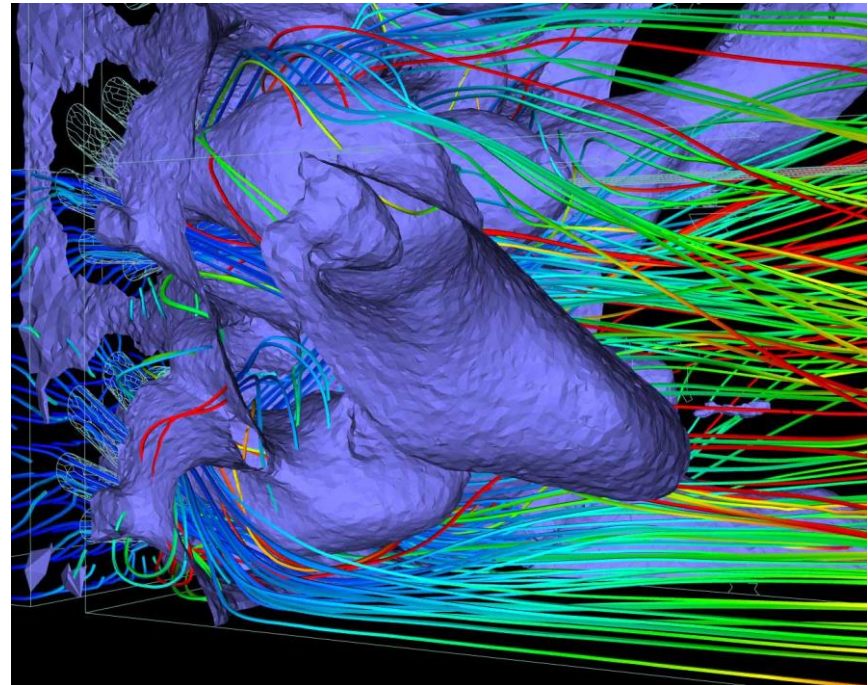
- Different types of 3D data vis
- Scientific visualization



en.wikipedia.org



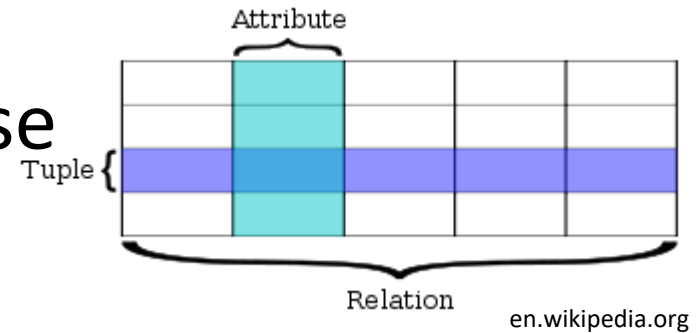
www.ornl.gov



gvis.grc.nasa.gov

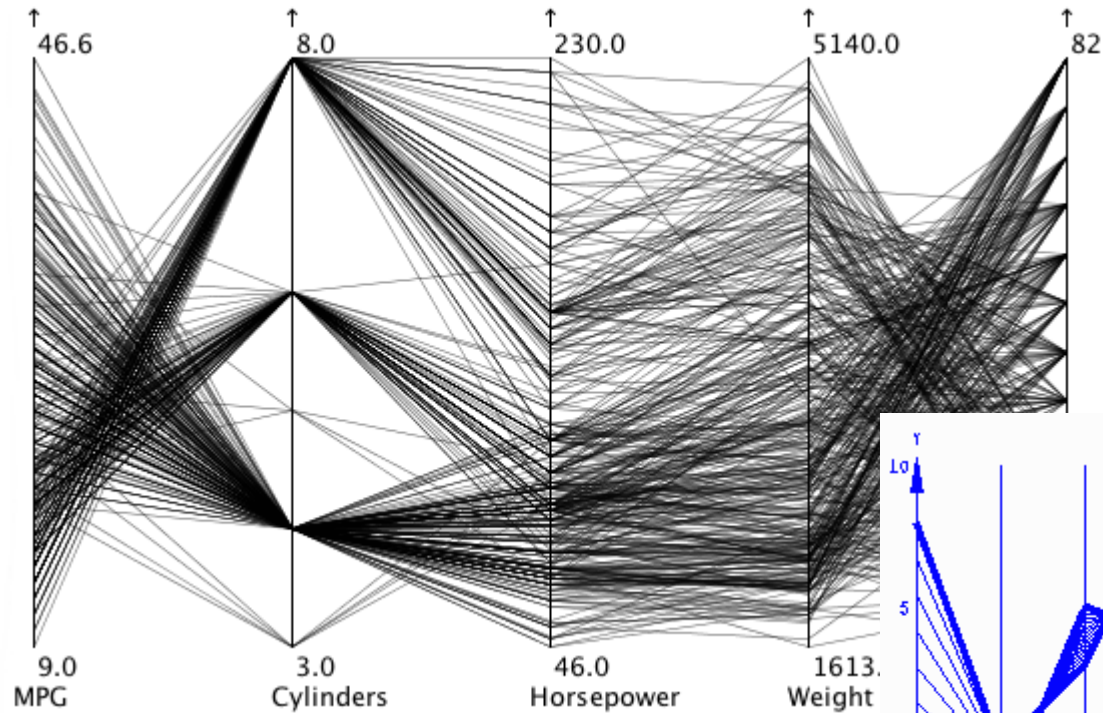
Multidimensional data

- Records in relational database

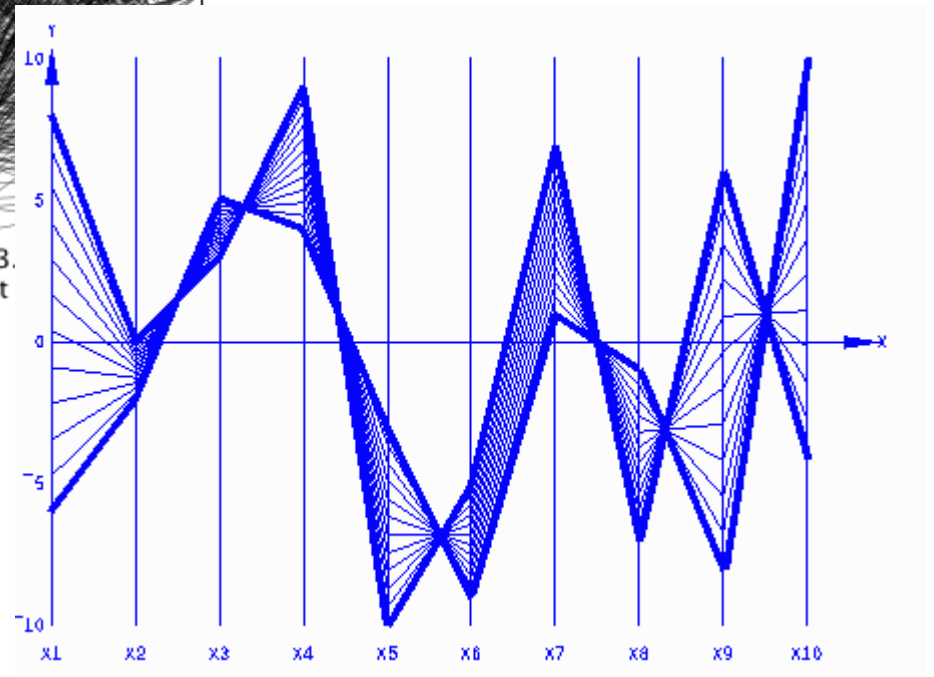


- Two solutions:
 - Drawing all possible pairs of variables in 2D graph
 - Simple but unusable for general overview of the data
 - „Parallel coordinates“
 - Method for displaying multidimensional data (Alfred Inselberg)

Parallel coordinates



eagereyes.org

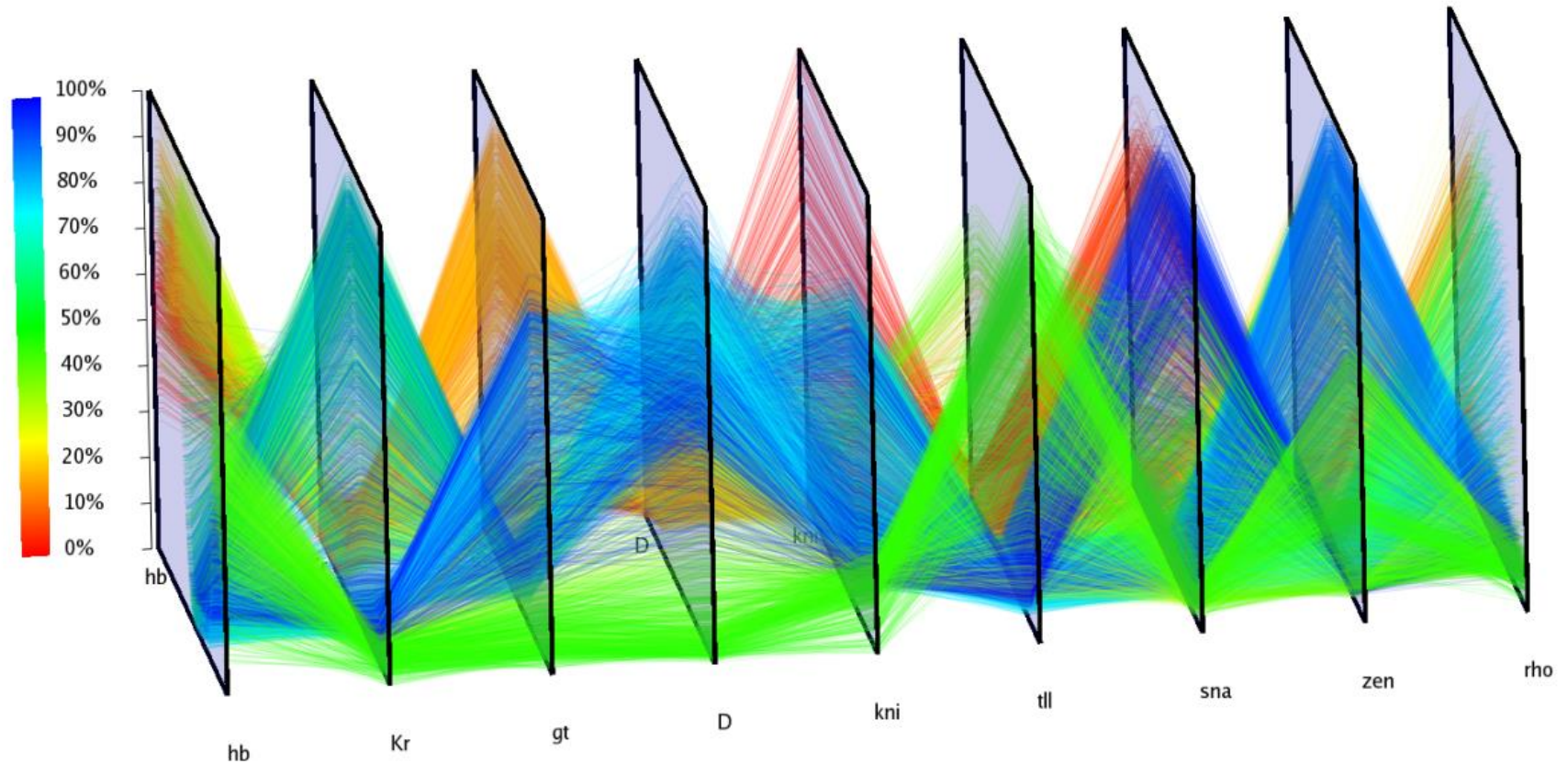


andrewgelman.com


Parallel coordinates

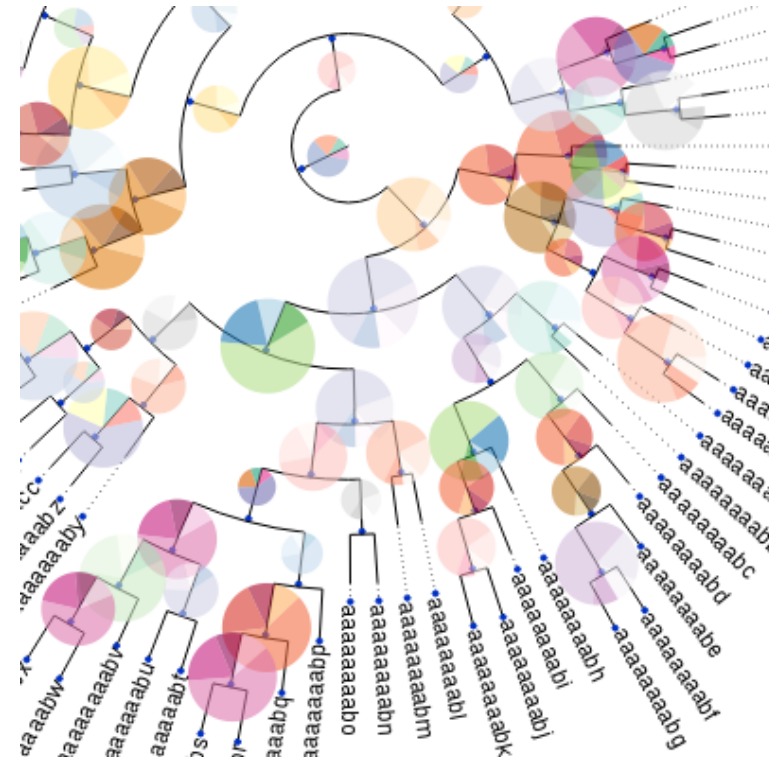


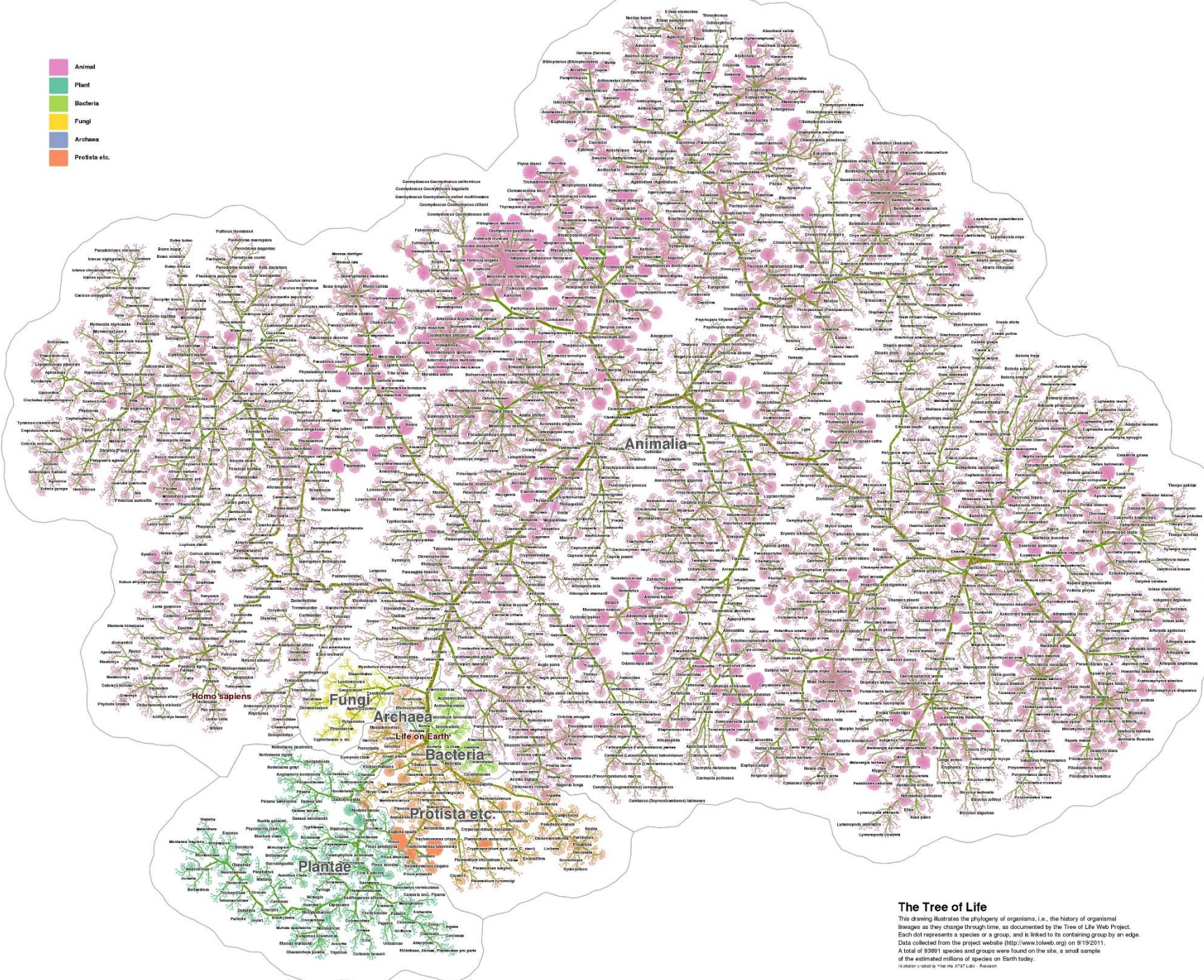
whyevolutionistrue.wordpress.com



Trees

- Displays not only data itself but also their structure
 - e.g., genetic trees, file systems
 - Number of items increases significantly in lower levels of tree
- 

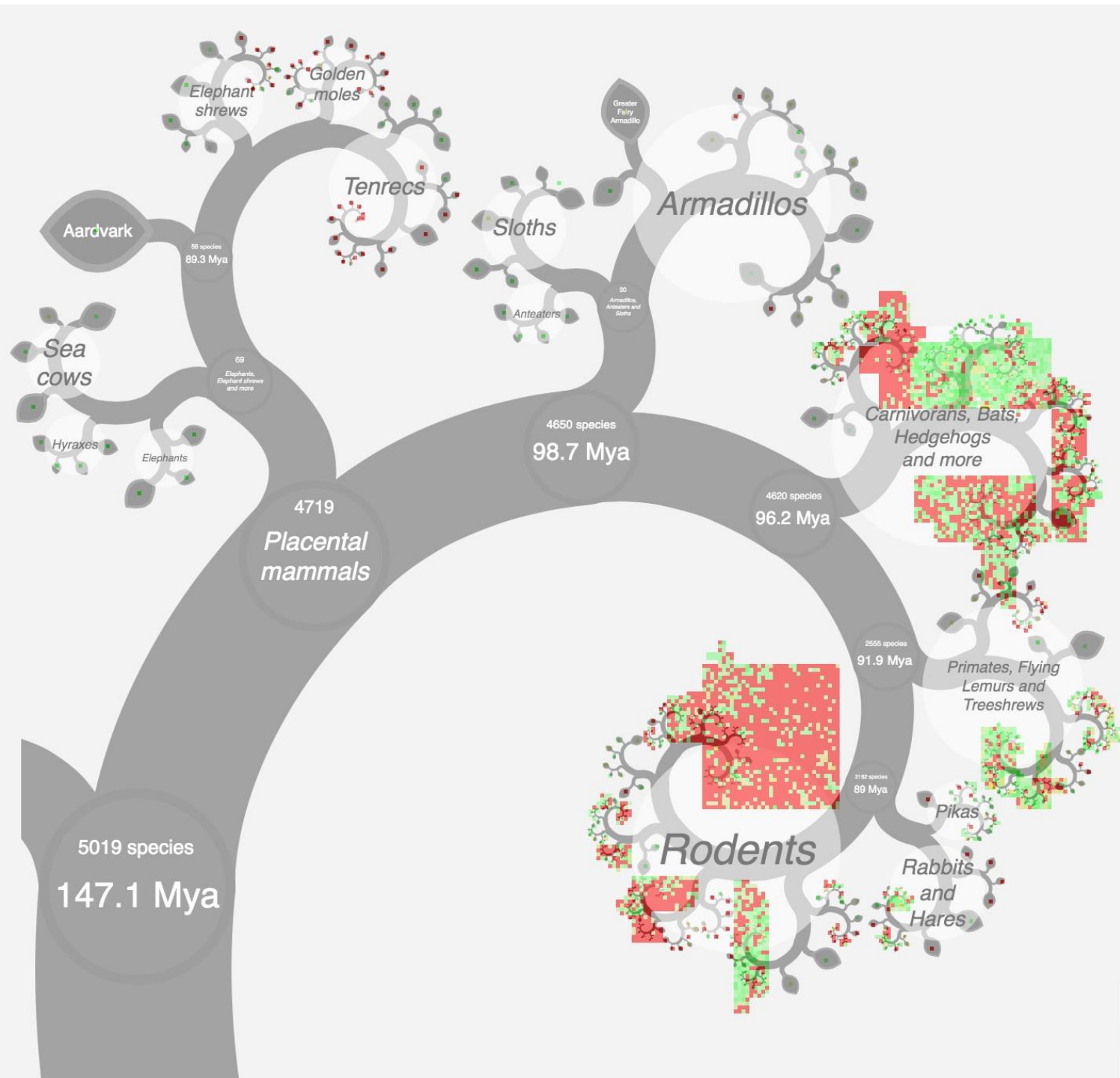




The Tree of Life

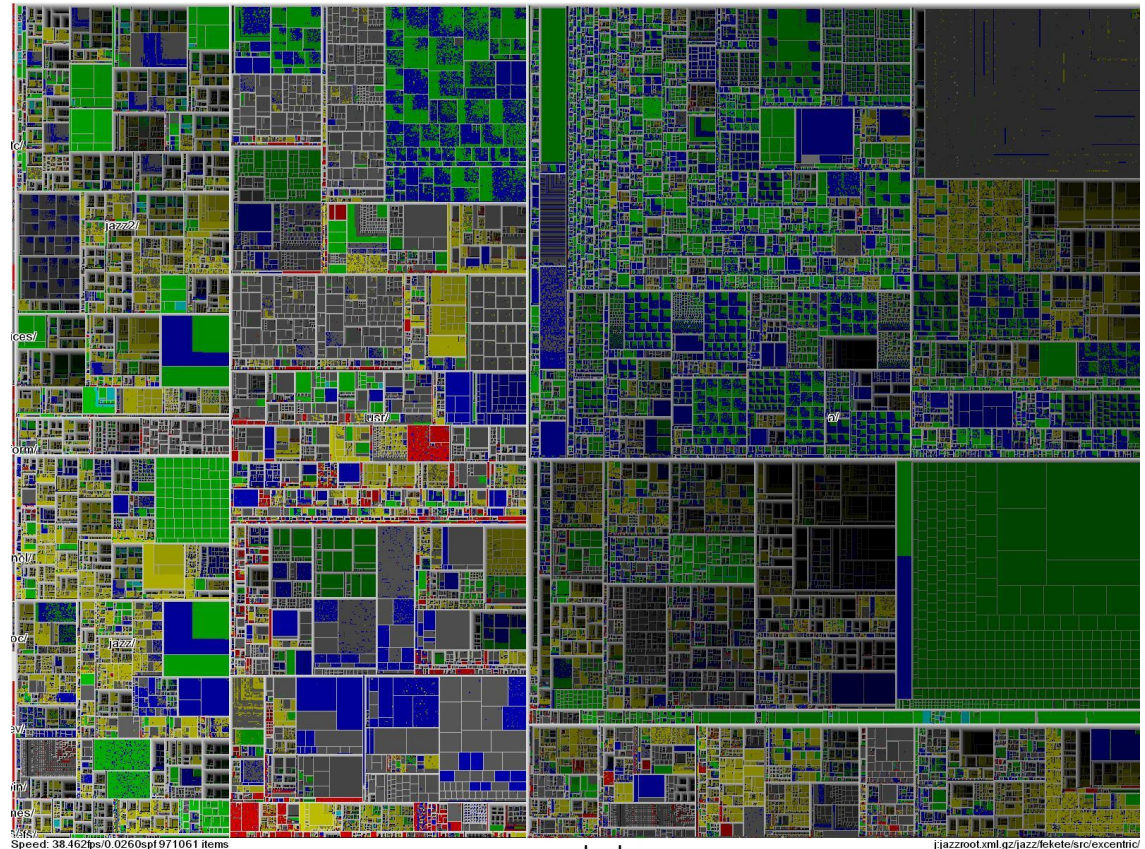
This drawing illustrates the phylogeny of organisms, i.e., the history of organismal lineages as they change through time, as documented by the Tree of Life Web Project. Each dot represents a species or a group, and is linked to its containing group by an edge. Data collected from the project website (<http://www.tolweb.org>) on 9/19/2011. A total of 93961 species and groups were found on the site, a small sample of the estimated millions of species on Earth today.

(Illustration created by Yael Paz, AT&T-LTES Research)



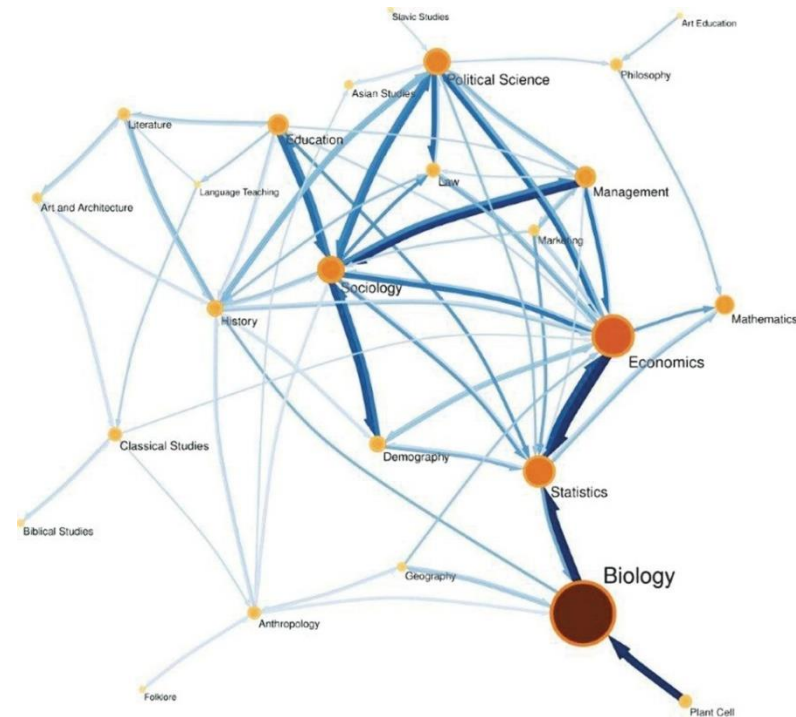
Trees

- Tree Maps
 - Displaying the tree data as nested rectangles
 - Tree with a million records:

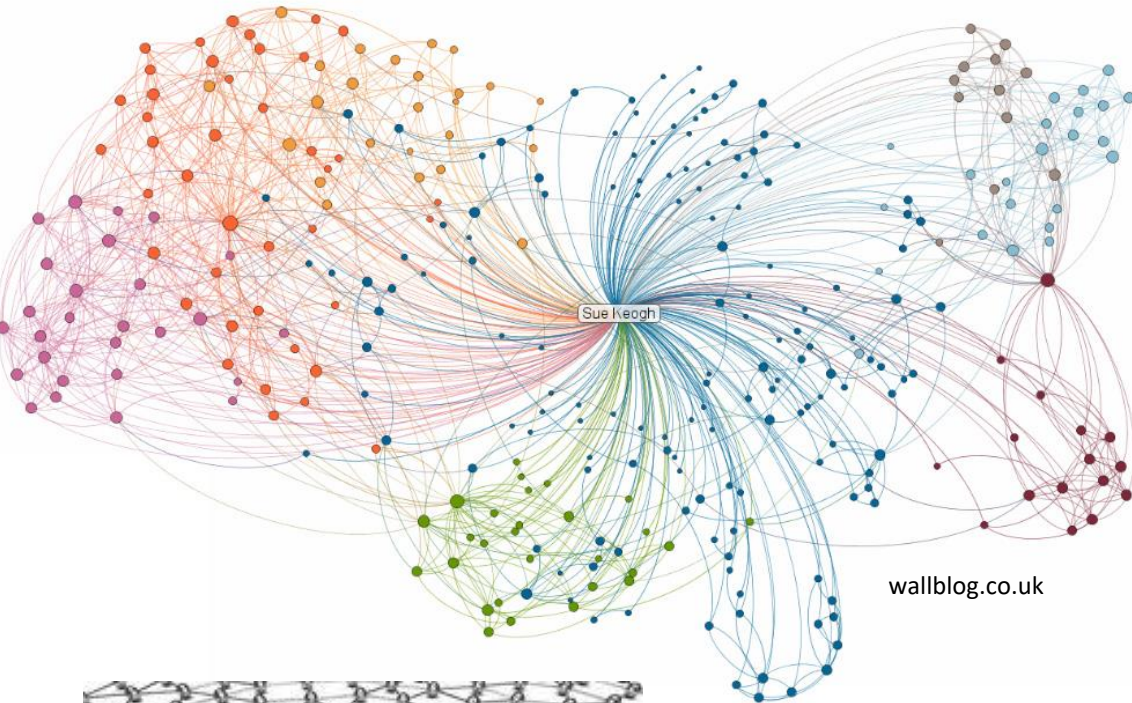


Networks

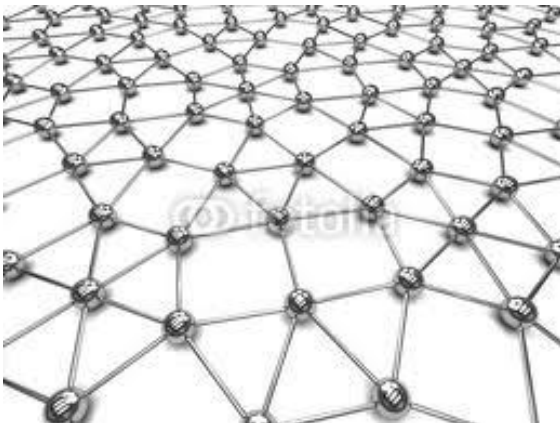
- Similar to trees – we aim to display the data structure
- Networks = nodes + edges
- Design should contain:
 - Minimal edge crossings
 - Minimal edge length
 - Minimal edge bending



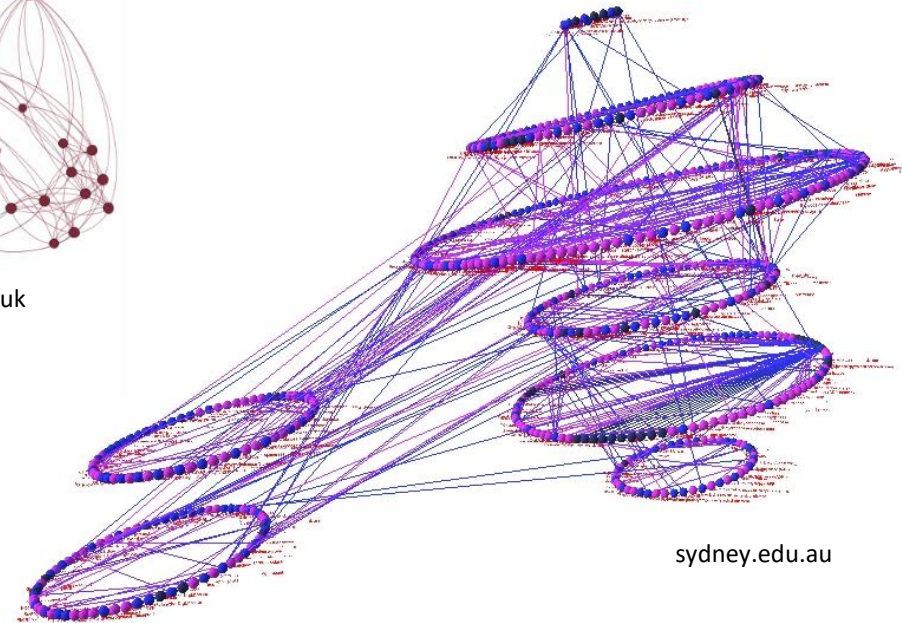
Networks



wallblog.co.uk



us.fotolia.com



sydney.edu.au

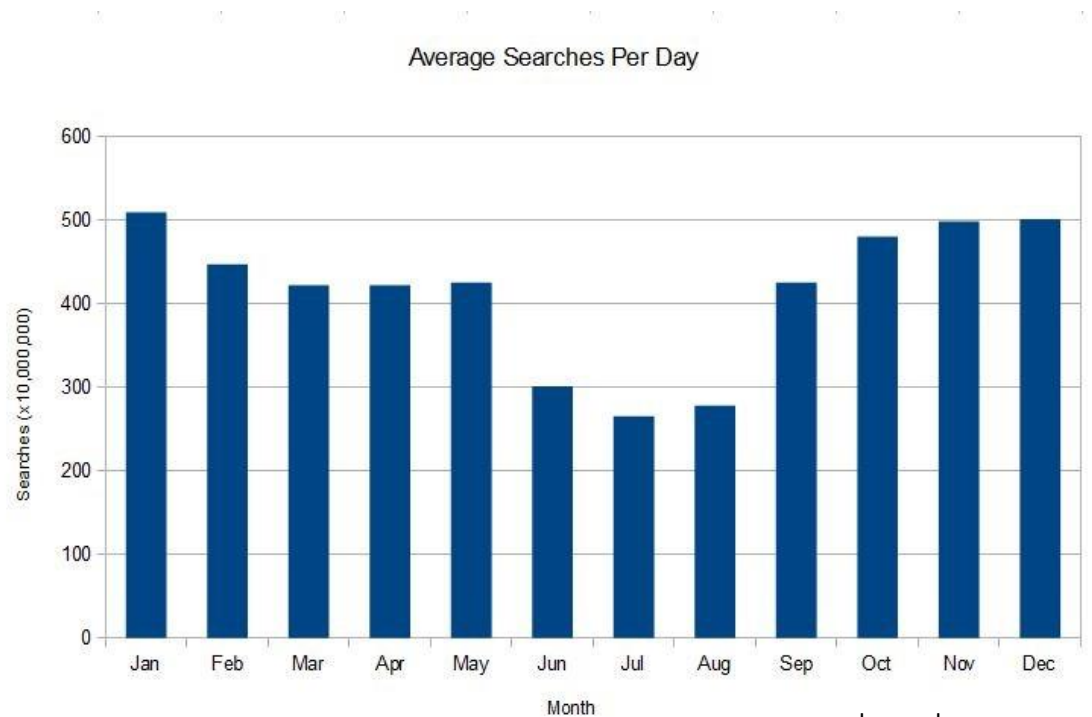
Temporal data

- Displaying data dependent on time
 - Trend and seasonal graphs

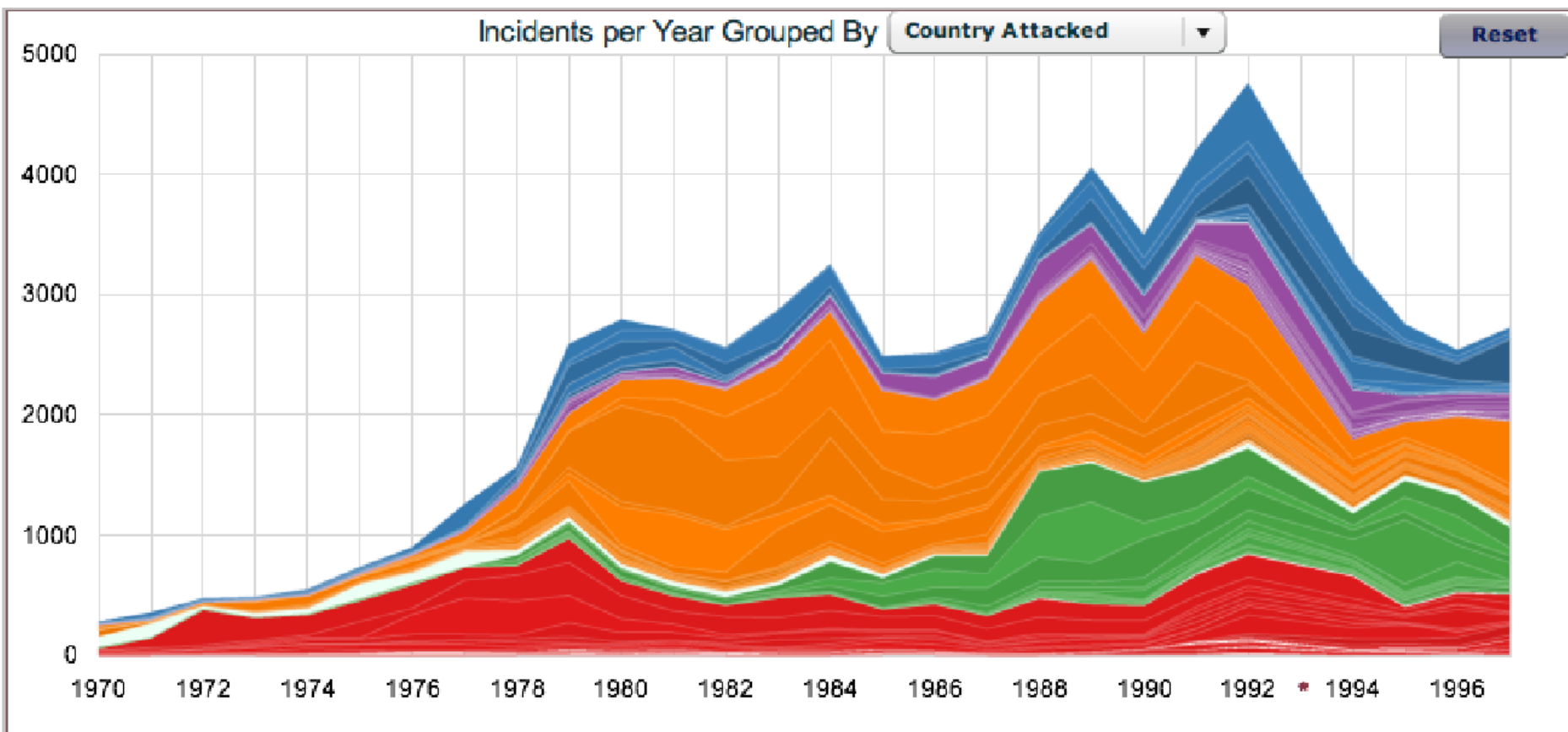


Resolution 3000 x 2200 px - Free JPG file download - www.psdgraphics.com

www.psdgraphics.com



www.demondeemon.com



Search

Rank of Incident Count

show between

1

203

max

min

Name

Colombia

Peru

El Salvador

Northern Ireland

Spain

India

Turkey

Sri Lanka

Chile

Count

5437

5136

4729

3382

2669

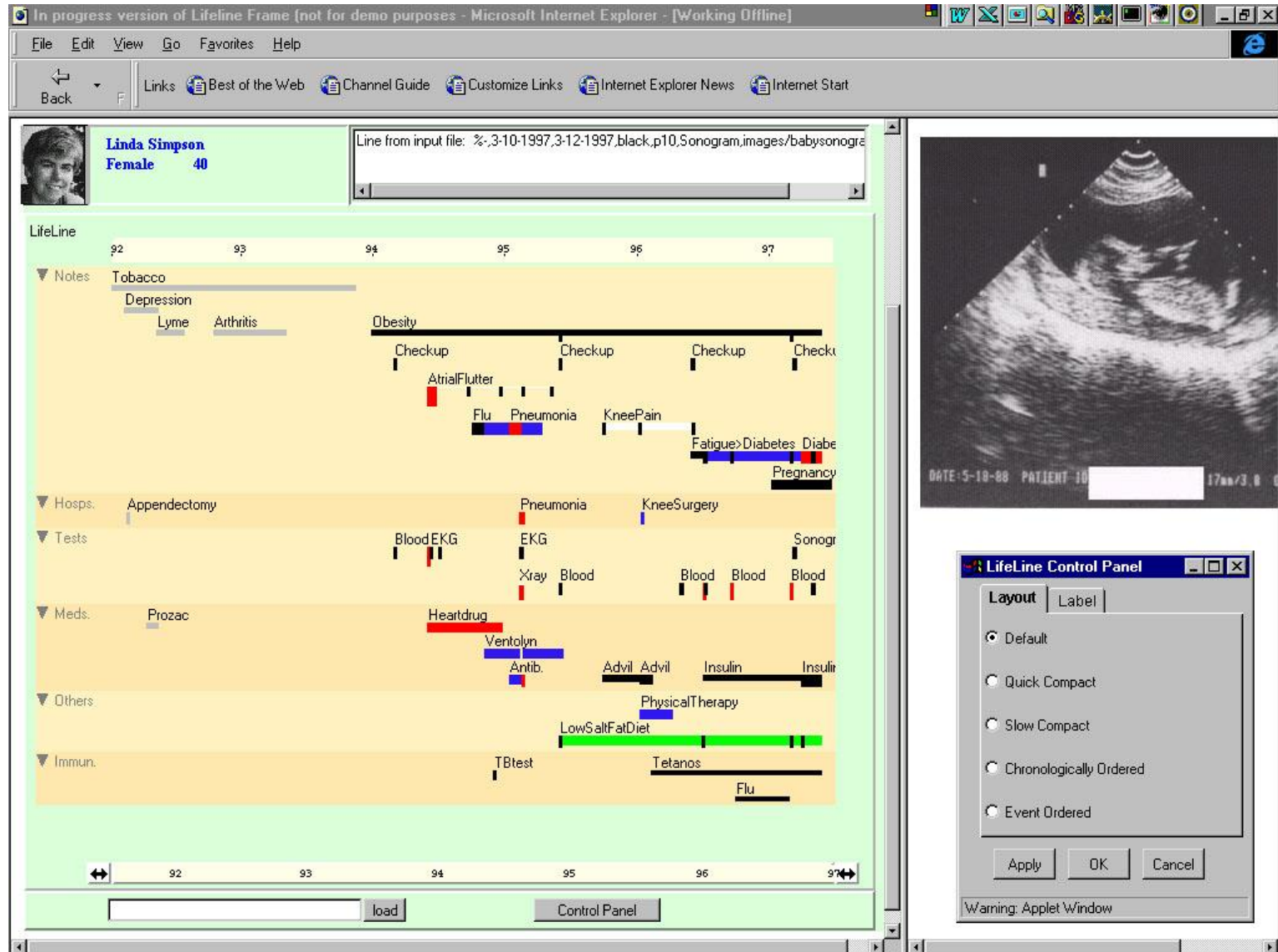
2648

2243

2010

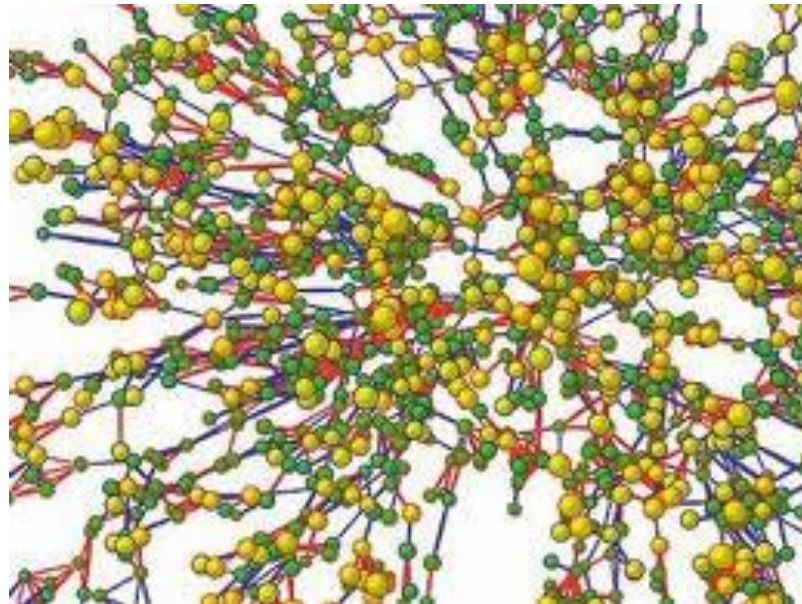
2004

Temporal data - LifeLines



Data preprocessing

- Displaying raw data = precise, identification of outliers, missing data, ...
- Sometimes preprocessing is required



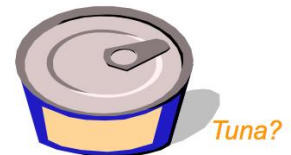
Preprocessing – techniques

- Metadata and statistics
- Missing values and data “cleaning”
- Normalization
- Segmentation
- Sampling and interpolation
- Dimension reduction
- Data aggregation
- Smoothing and filtration
- Raster to vector

Metadata and statistics

- Metadata – information for preprocessing
 - Reference point for measurement
 - Unit of measurement
 - Symbol for missing values
 - Resolution
- Statistical analysis
 - Detection of missing records
 - Cluster analysis
 - Correlation analysis

If you had two cans without labels, which would you eat?



Tuna?

Without a label, how would you know which was tuna and which was cat food?



Cat Food?

Missing values and data “cleaning”

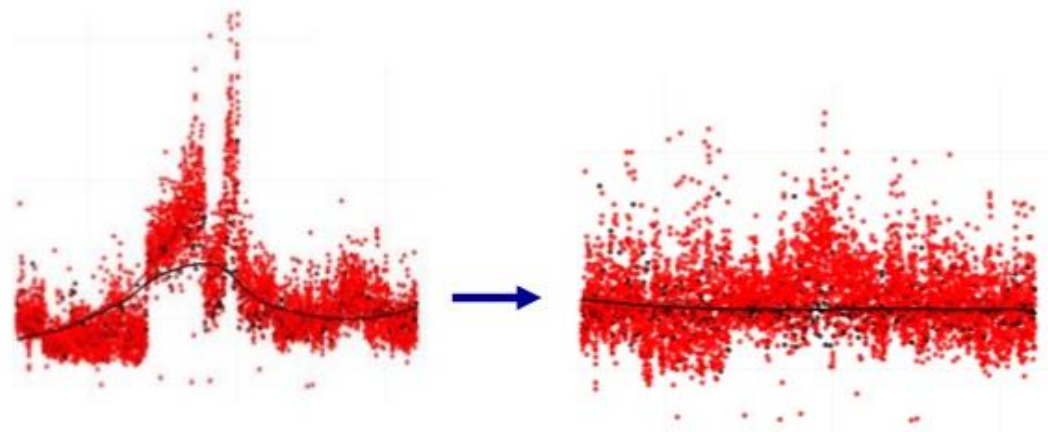
- Removing wrong records
- Assigning a given value
- Assigning an average value
- Assigning a value derived from the nearest neighbor value
- Calculating the value (imputation)

Normalization

- Transformation of the input dataset
- Adjusting values measured on different scales to a notionally common scale
- Normalization to interval [0.0, 1.0]:

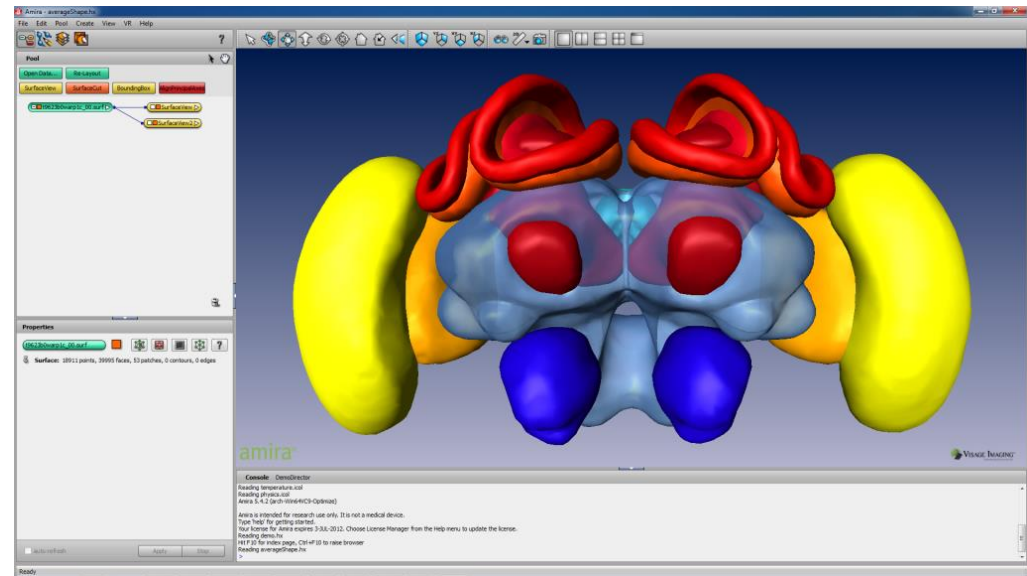
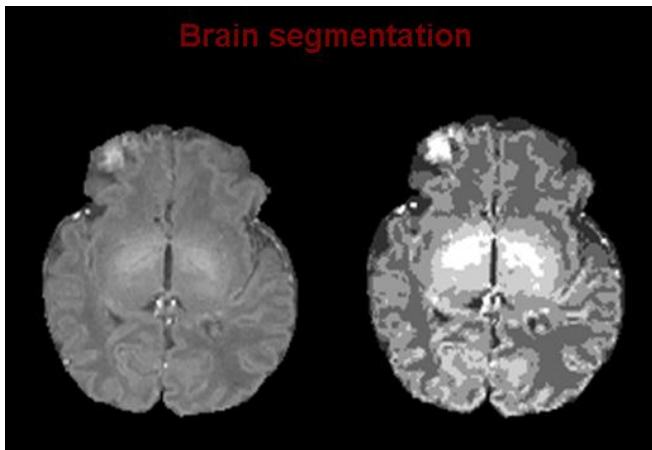
$$d_{\text{normalized}} = (d_{\text{original}} - d_{\text{min}}) / (d_{\text{max}} - d_{\text{min}})$$

- Clamping according to the threshold values



Segmentation

- Classification of input data into given categories
- Split-and-merge iterative algorithm



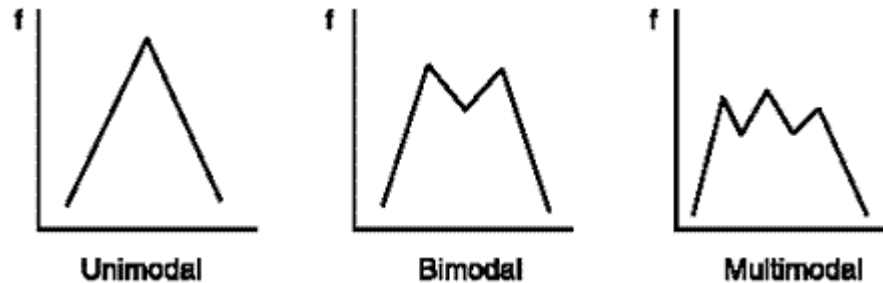
Split-and-merge

- similarThresh = defines the similarity of two regions with given characteristics
- homogeneousThresh = defines the region homogeneity (uniformity)

```
do {  
    changeCount = 0;  
    for each region {  
        compare region with neighboring ones and find the most similar one;  
        if the most similar one is within similarThresh of the current region {  
            connect these two regions;  
            changeCount++;  
        }  
        evaluate the homogeneity of the region;  
        if homogeneity of region is smaller than homogeneousThresh {  
            split the region to two parts;  
            changeCount++;  
        }  
    }  
} until changeCount == 0
```

Complex parts of the algorithm

- Determining the similarity of two regions
- Evaluating the homogeneity of a region – histogram



www.statcan.gc.ca

- Splitting the region

Possible problem

- Infinite loop by repeating split and merge steps of the same region
- Solution:
 - Changing the threshold value for similarity or homogeneity
 - Taking into account other region properties (e.g., size and shape of regions)

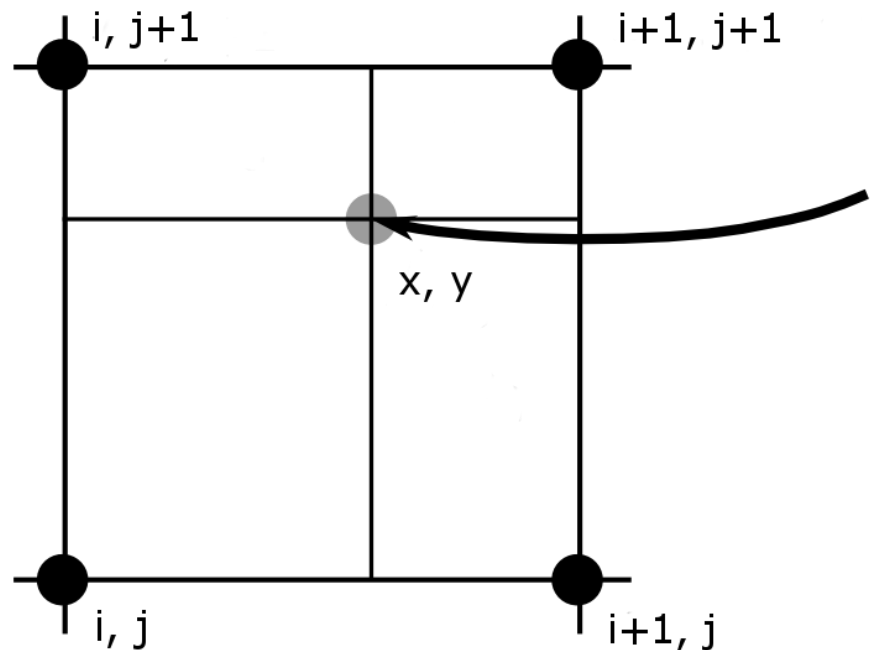
Sampling and interpolation

- Transformation of input data
- Interpolation = sampling method
 - Linear interpolation
 - Bilinear interpolation
 - Non-linear interpolation



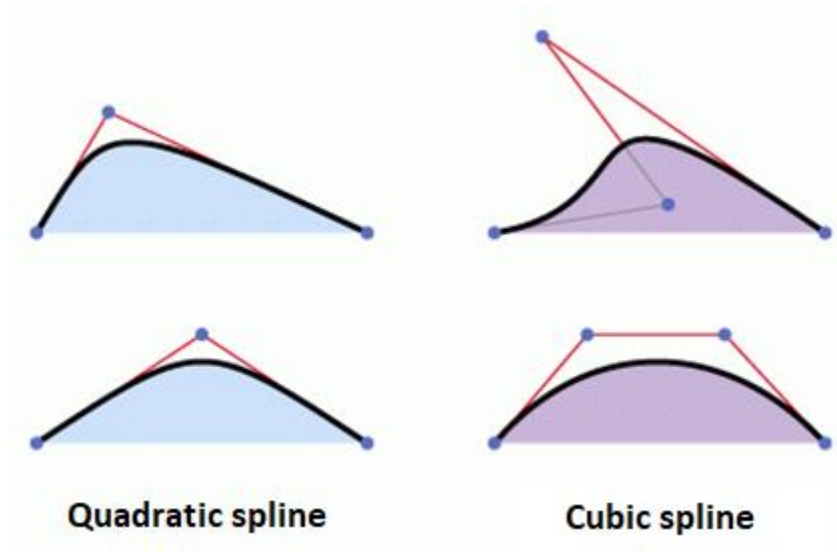
Bilinear interpolation

- Uniform grid
- Horizontal + vertical interpolation



Non-linear interpolation

- Problems with linear interpolation – zero connectivity in grid points
- Solution = using quadratic and cubic splines

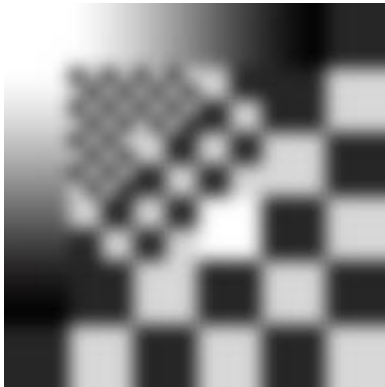


Result

- Original image (24x24 pixels)



cubic B-spline filter

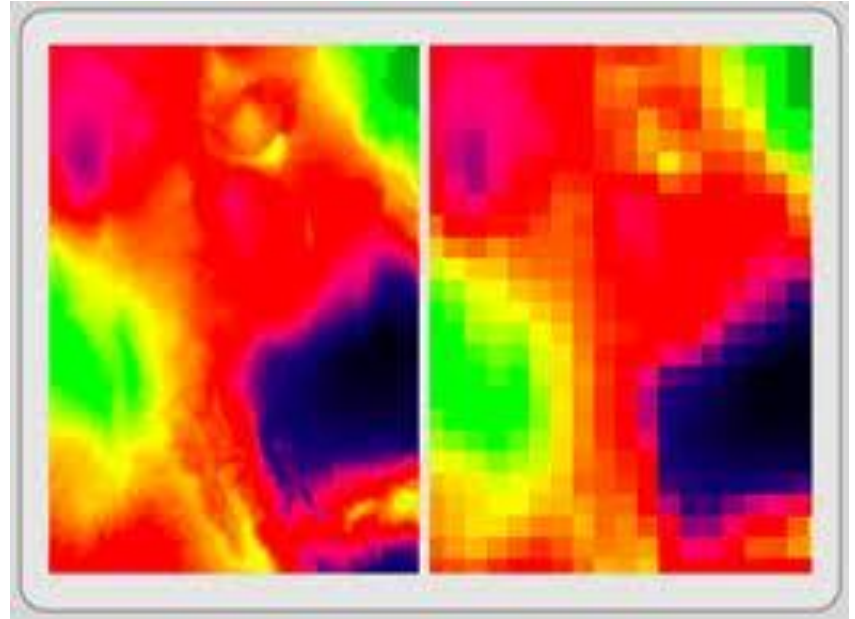


Catmull-Rom



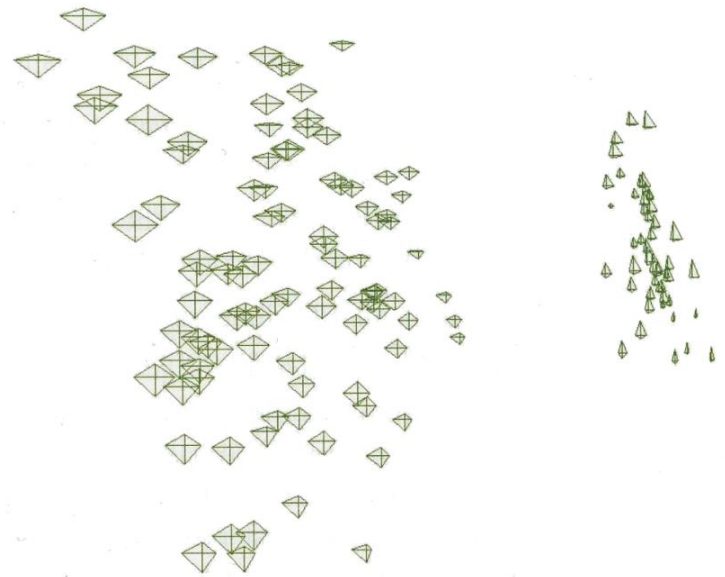
Resampling

- Pixel replication
- Neighbor averaging
- Data subsetting

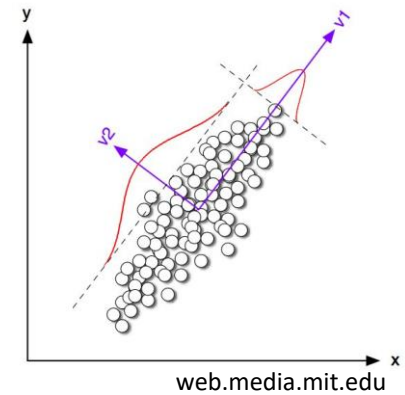


Dimension reduction

- Preparing multidimensional data for displaying
- Keep as much original information as possible
- Techniques:
 - **PCA** (principal component analysis)
 - **MDS** (multidimensional scaling)
 - **SOMs** (Kohonen self-organizing maps)



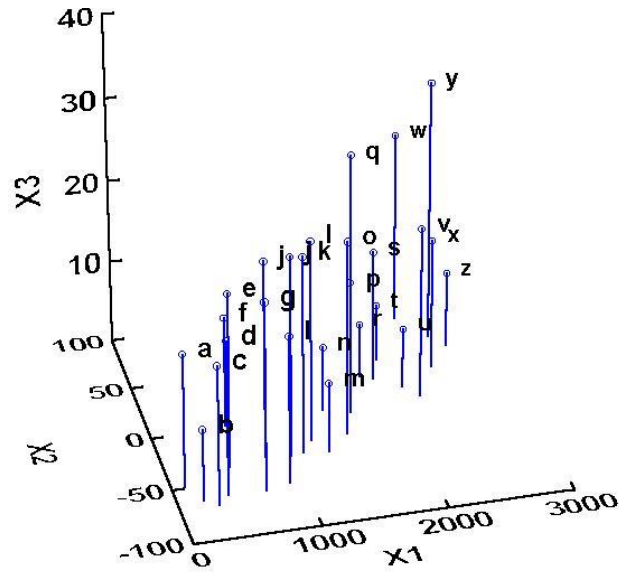
PCA intuitively



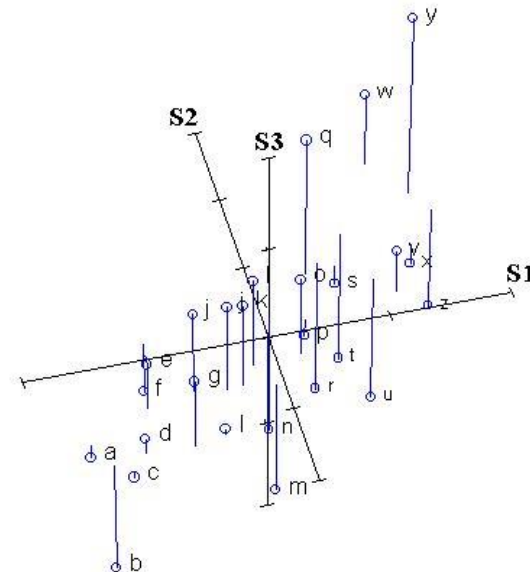
1. We select a line in space visualizing n -dimensional data. This line covers the most of the input data items and is called the first principal component (PC).
2. We select a second line perpendicular to the first PC, this forms the second PC.
3. We repeat this until we process all PC dimensions or until we reach a desired number of principal components.

PCA – principal component analysis

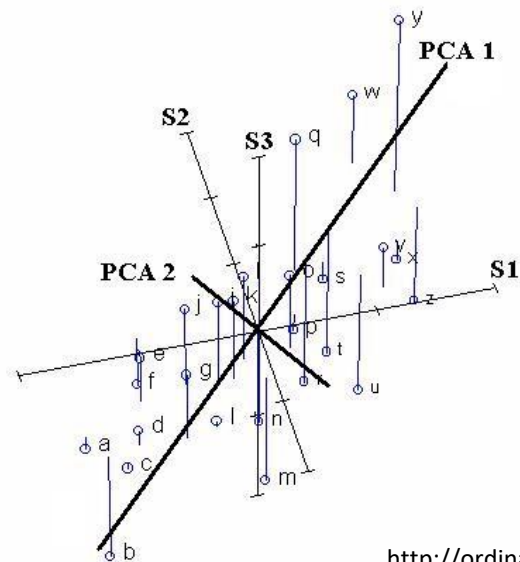
1)



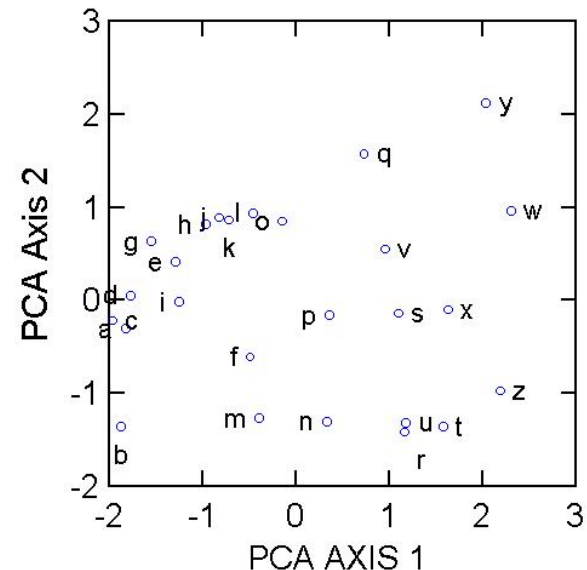
2)



3)

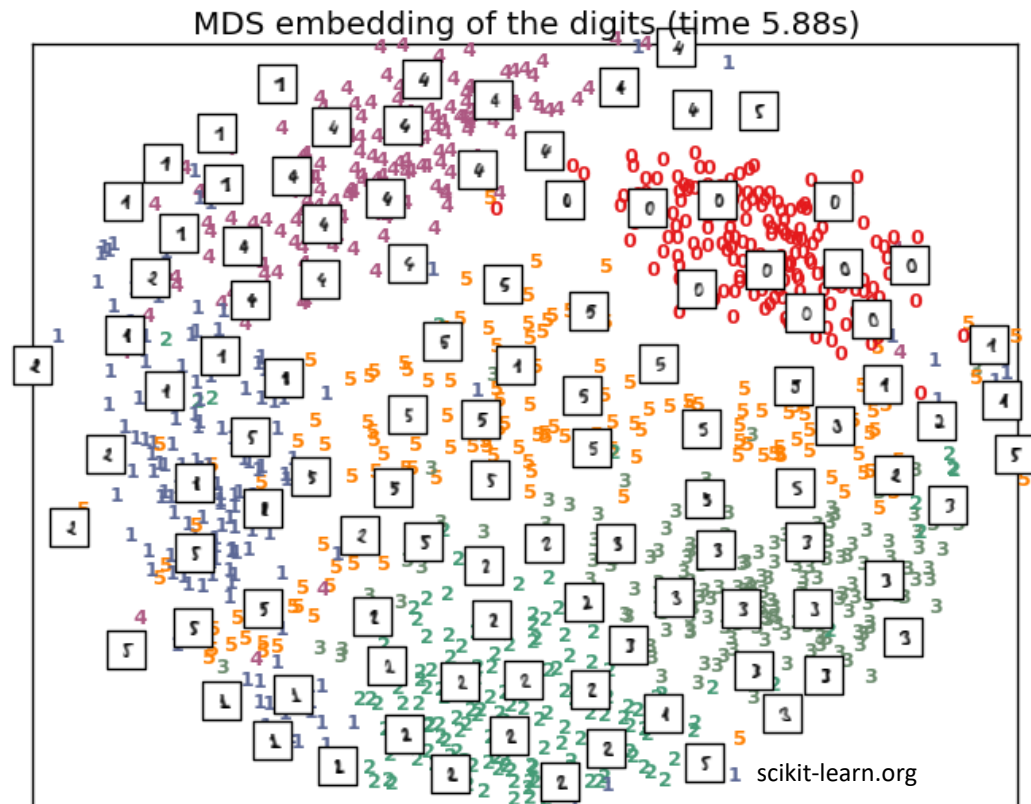


4)



MDS – multidimensional scaling

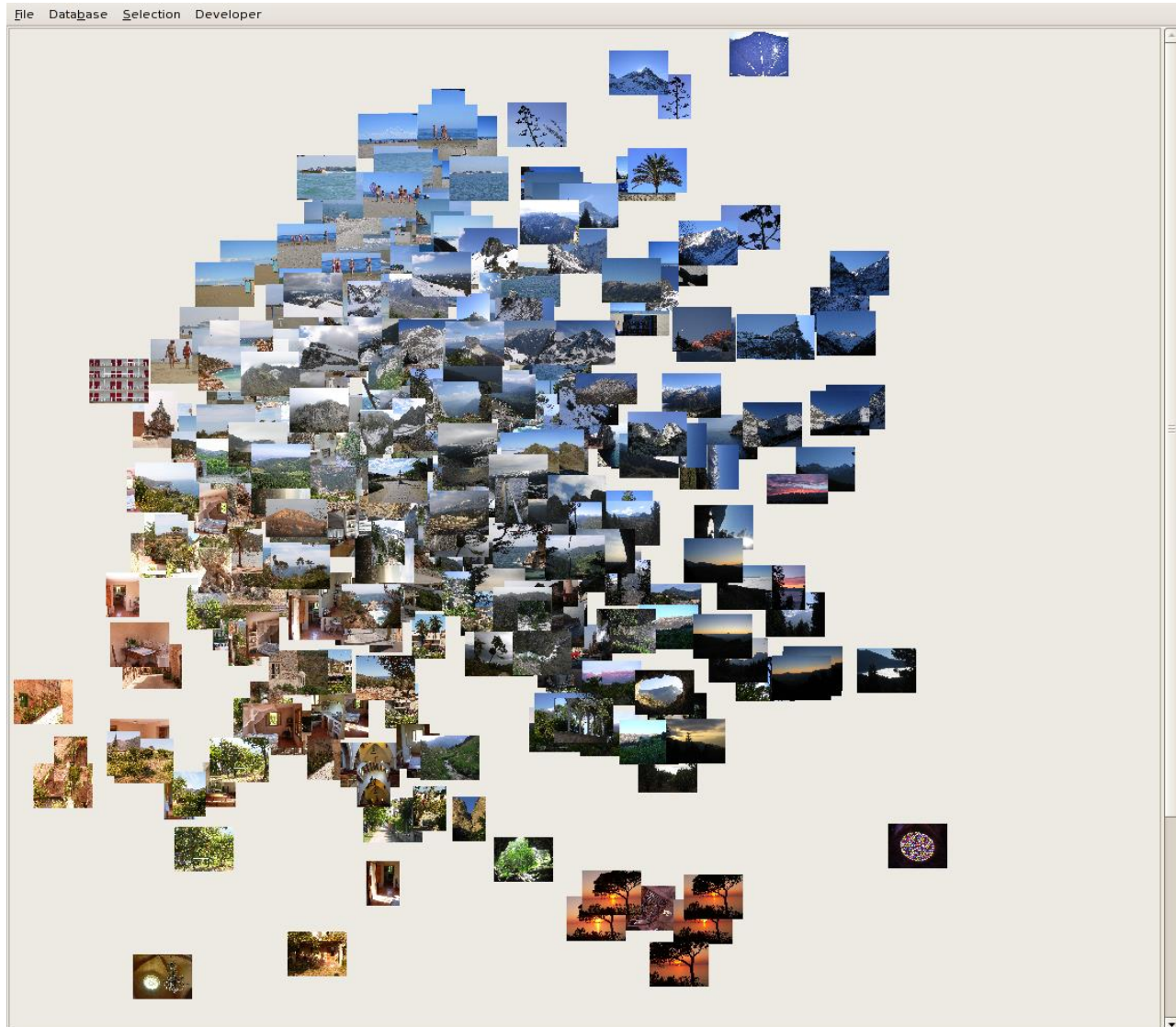
- Based on comparing the distances between individual data items in original and reduced space



MDS – multidimensional scaling

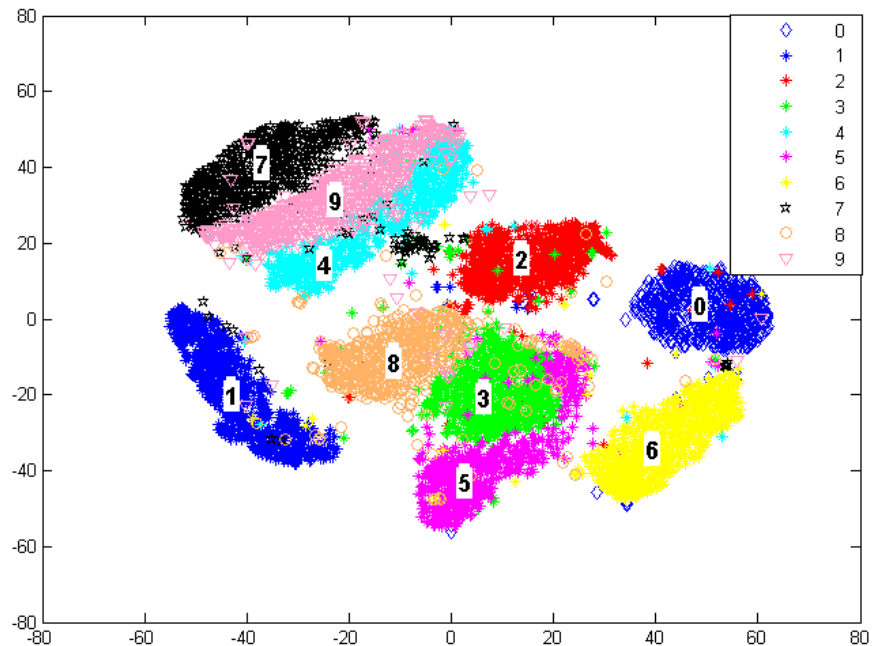
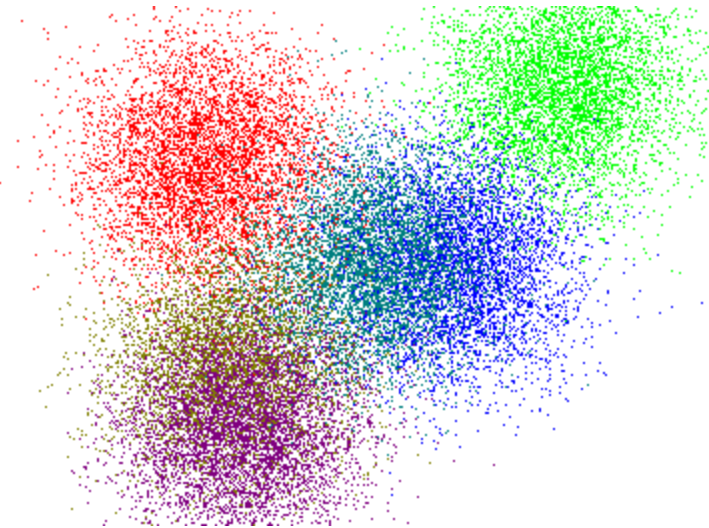
- 1) We calculate the distances between all pairs of data points in the original space. If we have n points as an input, this step requires $n(n - 1)/2$ operations.
- 2) We transfer all input data points to points in the reduced dimension space (often randomly).
- 3) We calculate *stress*, i.e., difference in distance between points in the original and reduced space. This can be done using different approaches.
- 4) If the average and cumulated *stress* value is smaller than the user-defined threshold, the algorithm ends and returns the result..
- 5) If the *stress* value is higher than the threshold, for each point we calculate a directional vector pointing to the desired shift direction in order to reduce *stress* between this point and the other points. This is determined as the weighted average of vectors between this point and its neighbors and its weight is derived from *stress* value calculated between individual pairs. Positive *stress* value repulses the points, negative one attracts them. The higher the absolute value of *stress*, the bigger movement of point.
- 6) Based on these calculations we transform the data points to the target reduced dimension, according to the calculated vectors. Return to step 3 of the algorithm.

MDS – multidimensional scaling



Data aggregation

- Aggregation = clustering of similar data to groups.



Smoothing and filtration

- Signal processing techniques – noise removal
- **Convolution** in 1D:

$$p_i = \frac{p_{i-1}}{4} + \frac{p_i}{2} + \frac{p_{i+1}}{4}$$

Converting rasters to vectors

- Used for:
 - Data compression
 - Image comparison
 - Data transformation
- Methods:
 - Thresholding
 - Region growing
 - Edge detection
 - ...



Conclusion

- The techniques mentioned improve the efficiency of visualization
- We have to inform the user that the data has been transformed!!!

