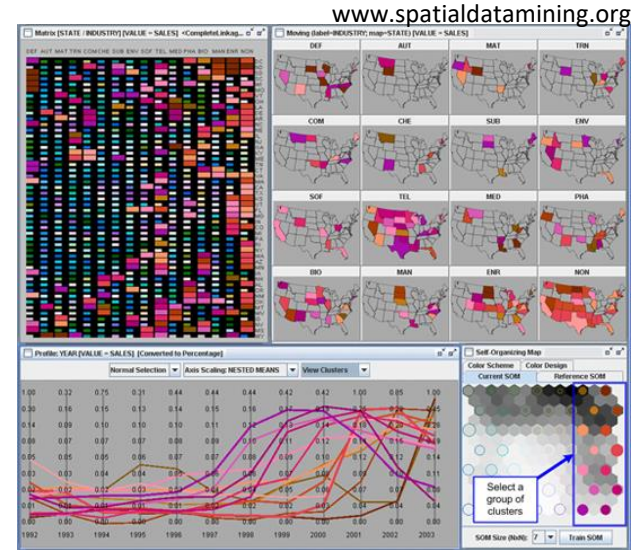
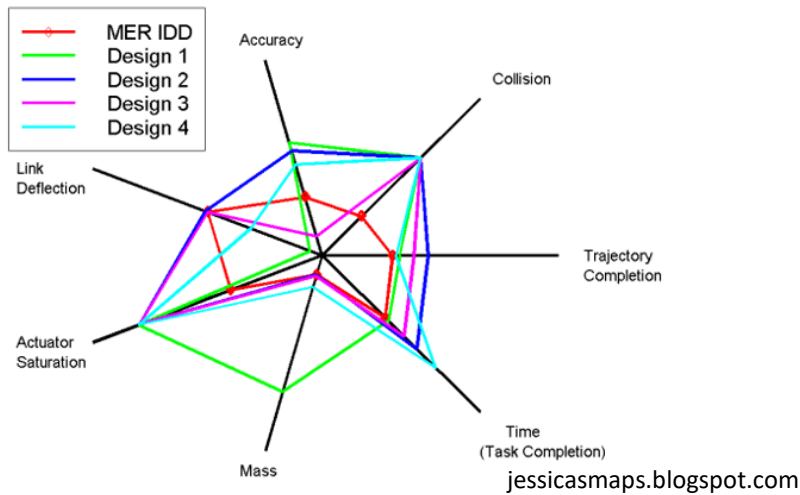
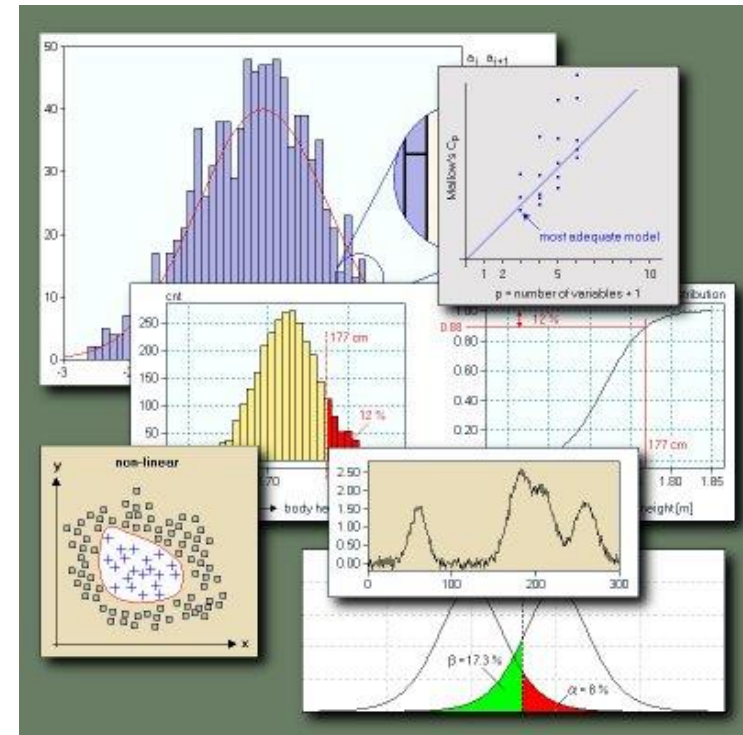
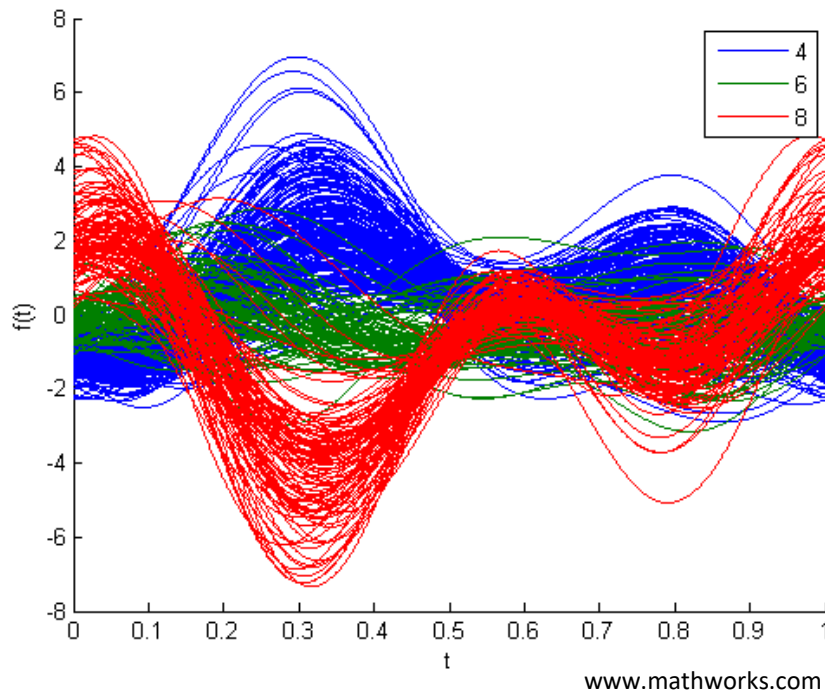


Star Plot of MER IDD and Automated Designs



5. Visualization of multivariate data



Multivariate Data

- Consist of multiple types of attributes
 - E.g., weight w , height h , shoe size s of randomly selected sample of people
 - The triples $(w_1, h_1, s_1), (w_2, h_2, s_2)$ then form a set of multivariate data
- Techniques for visualization of lists and tables of data that generally do not contain explicit spatial attributes

Point-Based Techniques

- Scatterplots - projection of data records from n -dimensional data space to an arbitrary k -dimensional space of output device
- Data records are mapped onto k -dimensional points
- Each record is associated with a certain graphical representation

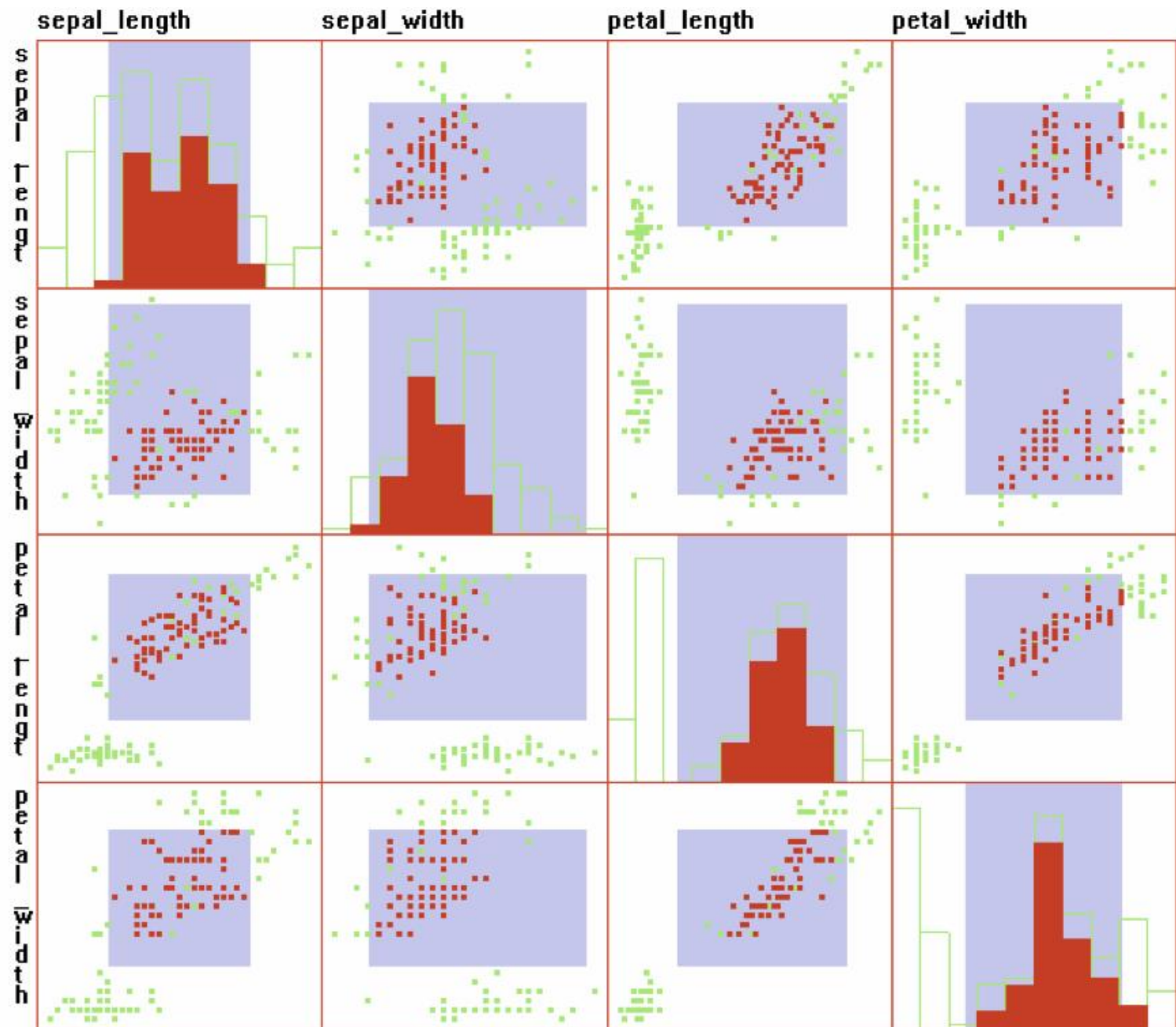
Scatterplots

- Data analysis consists of:
 1. Search for a subset of input data dimensions
 2. Dimension reduction (PCA, multidimensional scaling)
 3. Dimension embedding – mapping dimensions onto additional graphical attributes (color, size, shape)
 4. Multiple displays – displaying multiple plots together at once (superimposition, juxtaposition)

Multiple Displays

- **Scatterplot matrix**

- Grid containing scatterplots
- N^2 cells, where N is the number of dimensions
- Each dimension pair is displayed twice – just rotated by 90°
- Usually symmetric along the main diagonal
- The main diagonal can display
 - Description of corresponding dimension or
 - Histogram of the given dimension

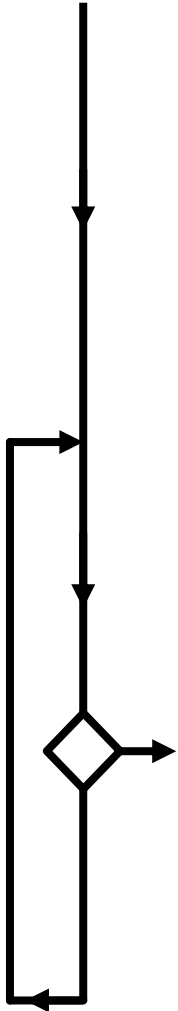


Force-Based Methods

- Projection of points from large dimensions into 2D or 3D space
- Aims to preserve the properties of N-dimensional data while projecting to different dimension
- Projection can introduce unwanted artifacts to appear in the resulting visualization

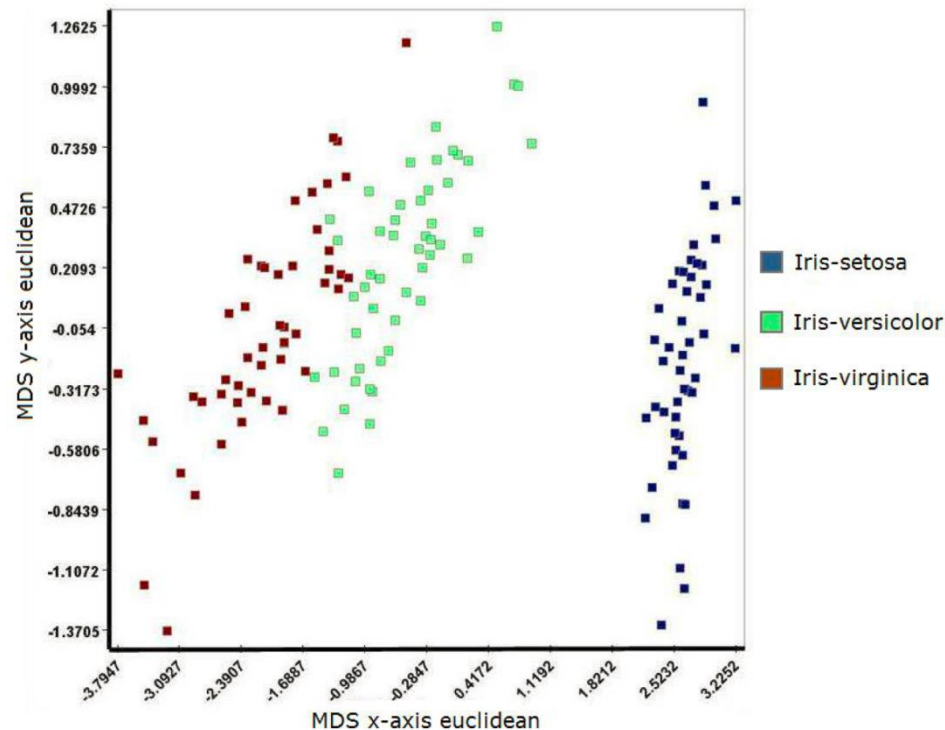
Multidimensional Scaling (MDS)

1. Given dataset consisting of M records and N dimensions, create $M \times M$ matrix D_s containing results of similarity measurement between individual record pairs
2. Supposing that we want to project the input data to K dimensions, create $M \times K$ matrix L , which contains placement of projected points
3. Compute $M \times M$ matrix L_s containing similarity between all record pairs from L
4. Compute the value *stress* S by measuring the differences between D_s and L_s
5. If S is sufficiently small, terminate the algorithm.
6. Else shift the positions of records in L in the direction which will reduce the *stress* value
7. Return to step 3



Multidimensional Scaling (MDS)

- Numerous variants of the algorithm exist. The main differences are in:
 - Method for similarity and stress computation
 - Definition of start and end conditions
 - Strategy for updating the position of points

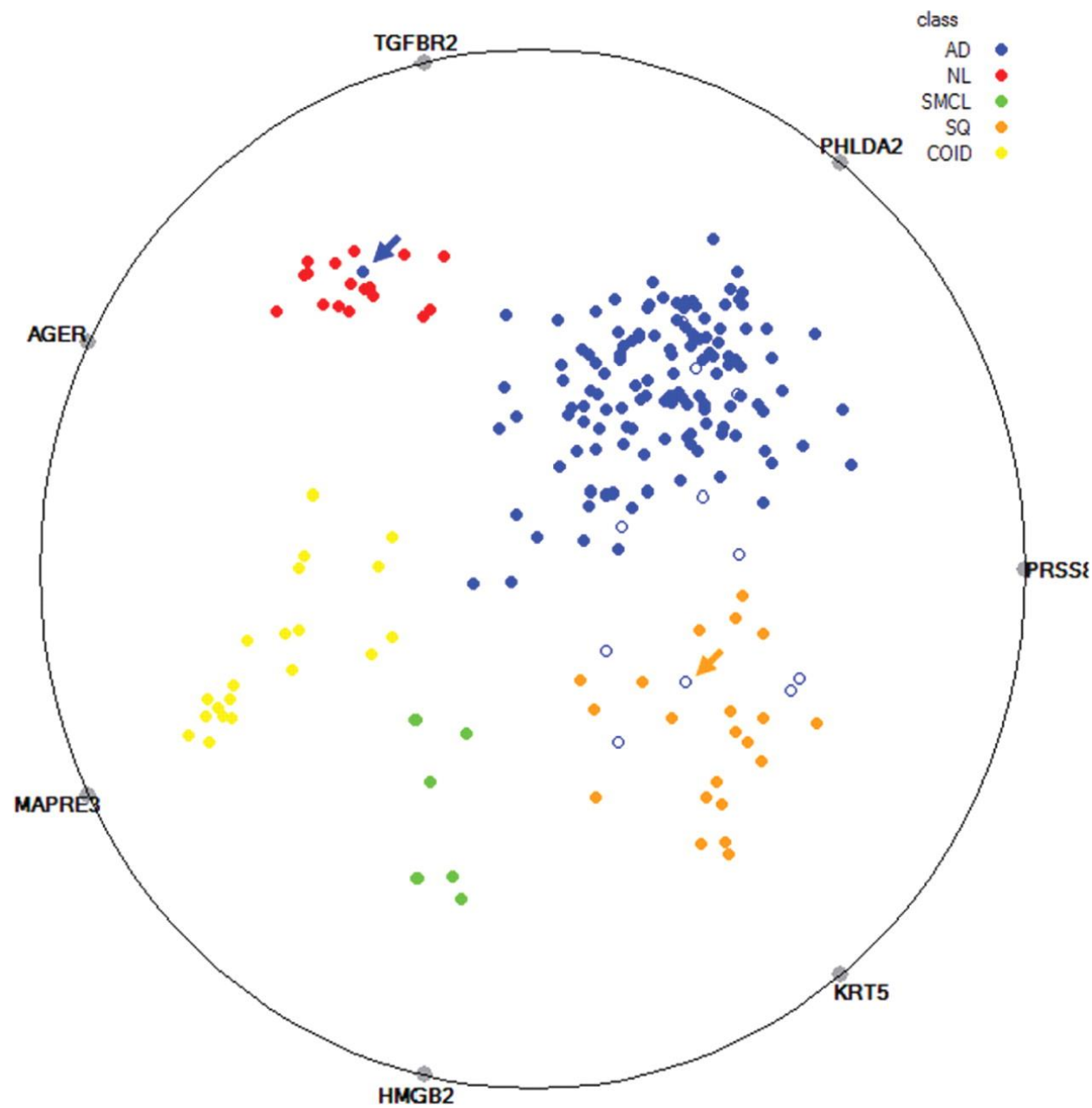


Problems

- Results are not unique – small changes in start conditions can lead to different results
- Coordinate system after the projection may not be easily understandable to the user – with respect to the dimensions of the original data
 - The most significant are the relative positions of individual points, rather than their absolute positions, which may differ from algorithm to algorithm

RadViz

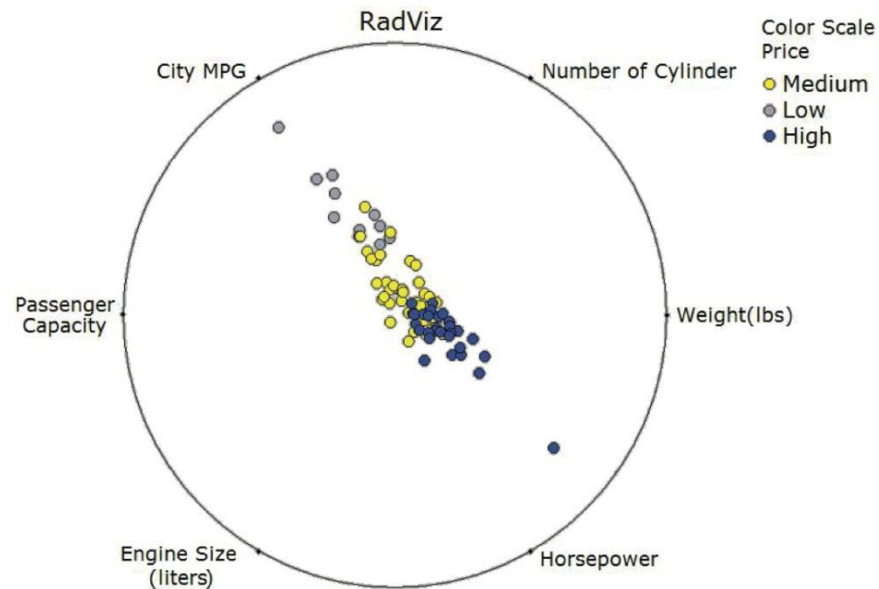
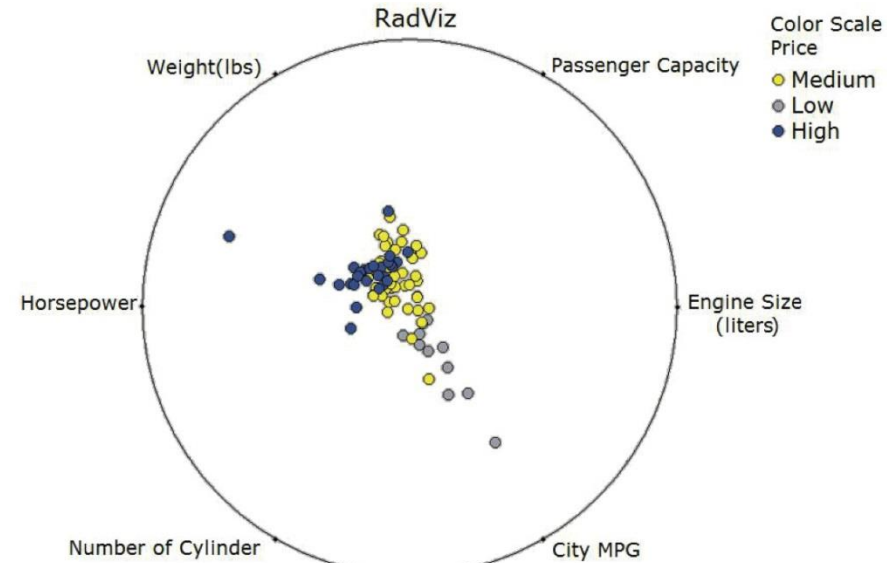
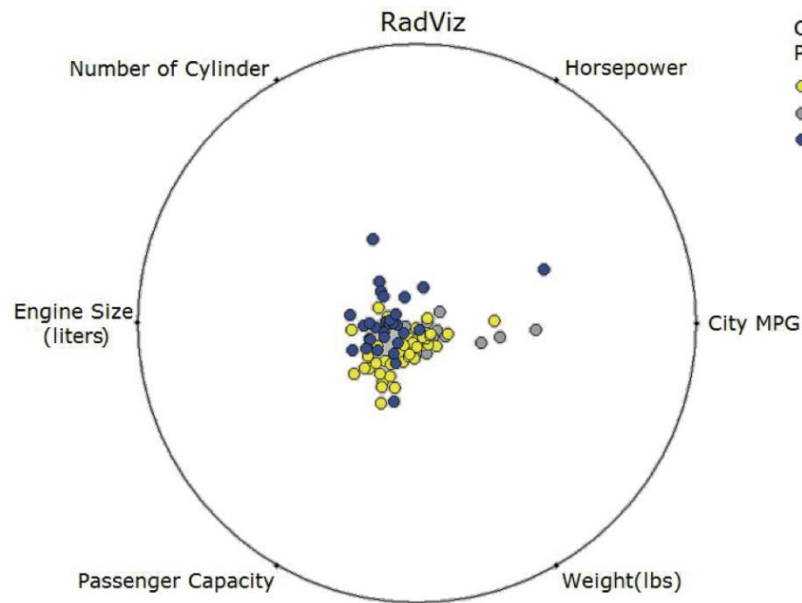
- Based on Hooke's law of elasticity for finding equilibrium position of the point.
- For N-dimensional dataset, N so-called “anchor” points are placed on the circumference of a circle (for simplicity we consider a unit circle placed at the origin of the coordinate system) – these represent fixed ends of N strings assigned to each data point.



RadViz

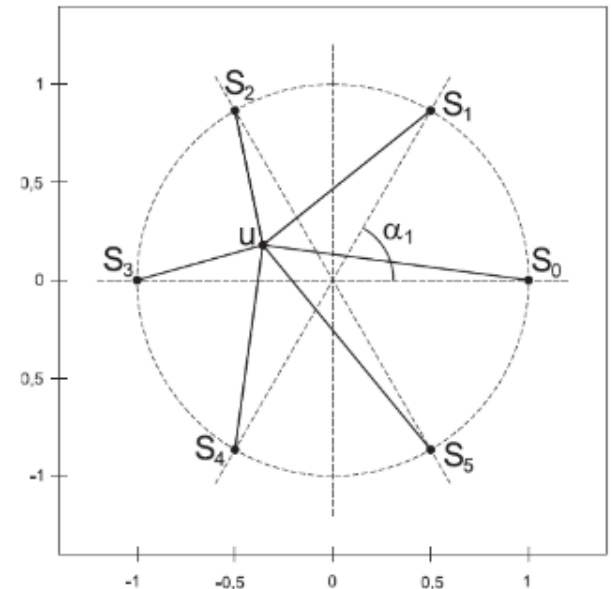
- Different placement and order of anchor points leads to different results
- Points with different position in the N-dimensional space can be mapped to the same position in 2D space
- These problems concern all the techniques for projection and dimension reduction
- The simple solution for RadViz is enabling the user to interact (manipulate) with anchor points

RadViz



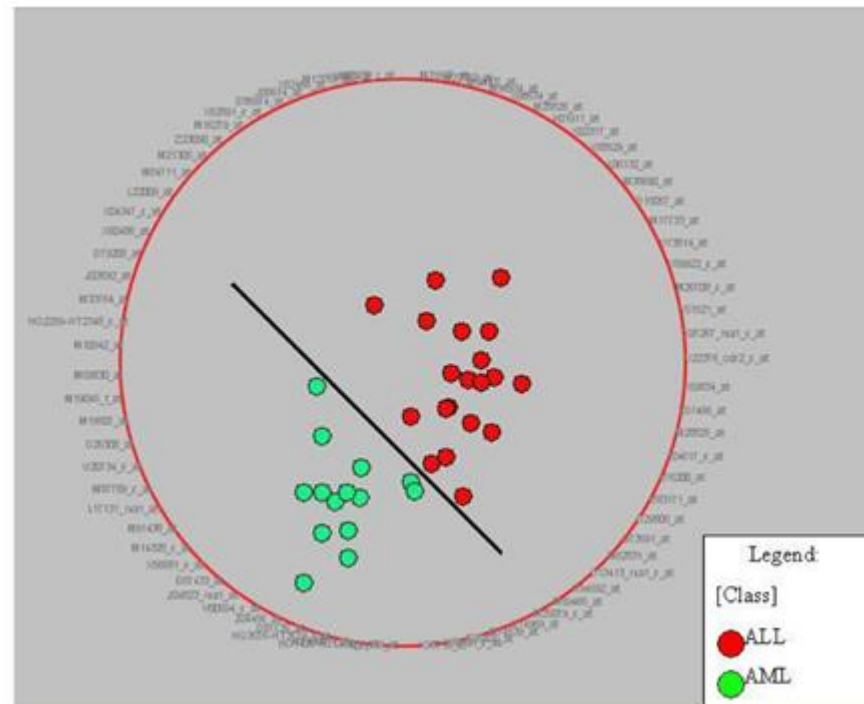
RadViz – analogical definition

- Point in N-dimensional space $[y_1, y_2, \dots, y_n]$
- To each anchor point S_j there is attached a virtual spring of rigidity y_j – changing according to the value of the given parameter
- All springs are connected at one point u
- We search for the equilibrium of the spring system



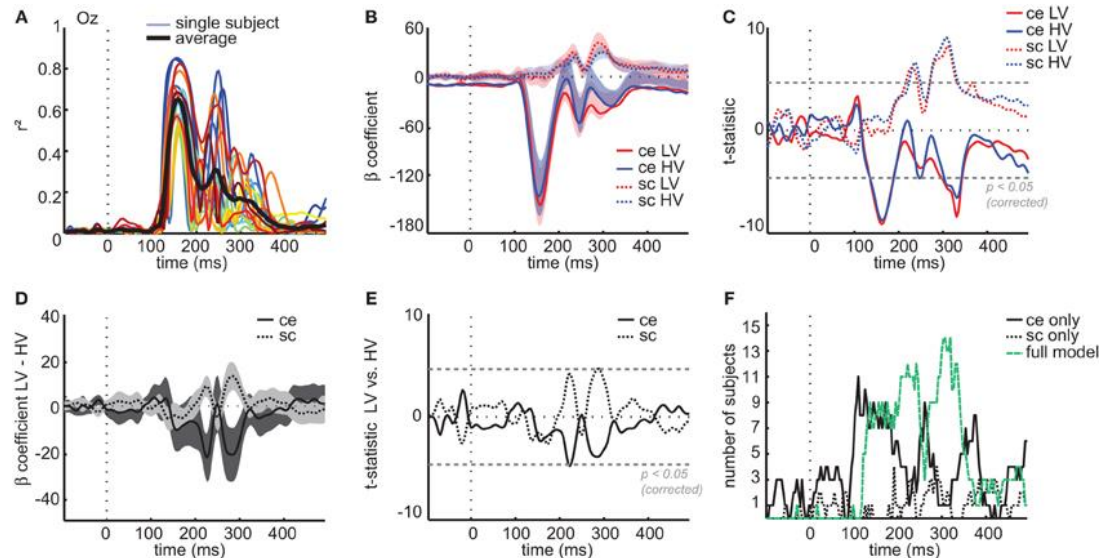
RadViz

- Algorithm searching for the arrangement of dimensions on the circumference of the circle leading to maximal dispersal of the data



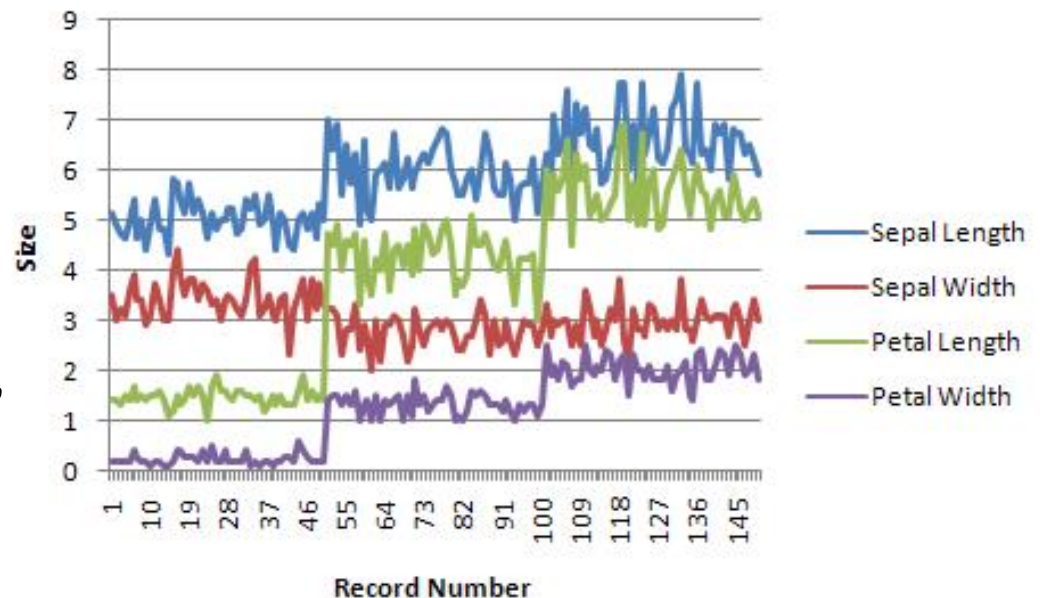
Line-Based Techniques

- Records are displayed in such way, that the corresponding points are connected with either straight or curved line
- Using additional properties, such as curvature, crossings, etc., lines can display relationships between data



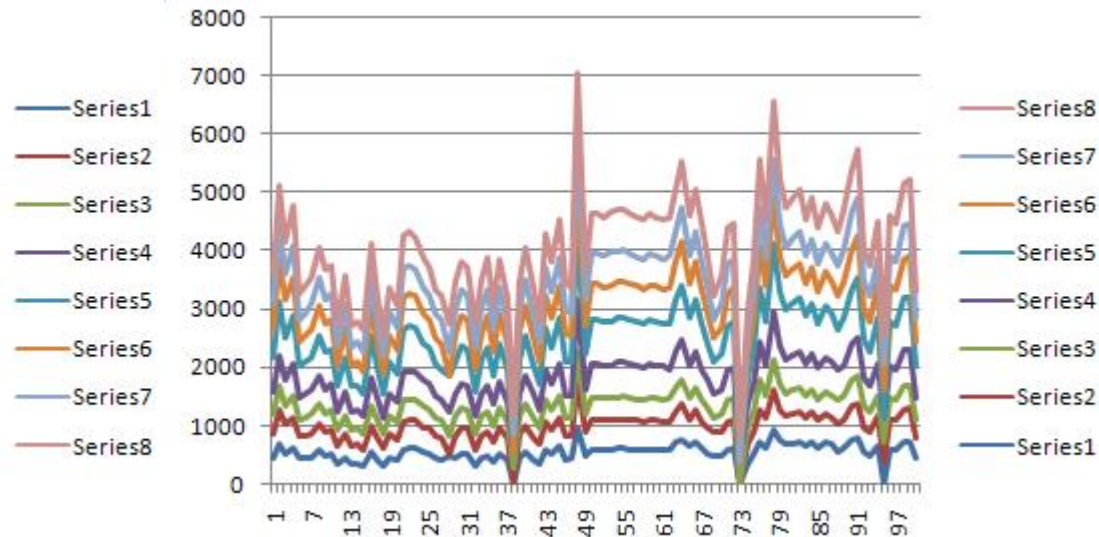
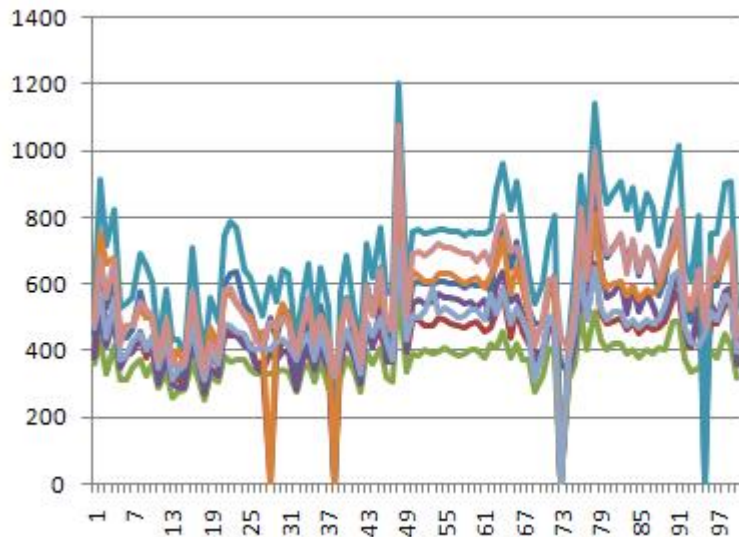
Line Charts

- Visualization technique for single variable, where vertical axis represents possible range of variable values and horizontal axis represents certain ordering of records in a given dataset
- Extension for multivariate data
 - superimposition,
 - juxtaposition



Line Charts

- Classic line chart for 8-dimensional dataset vs. stacked line chart (for each added dimension the chart of previous dimension serves as the base)



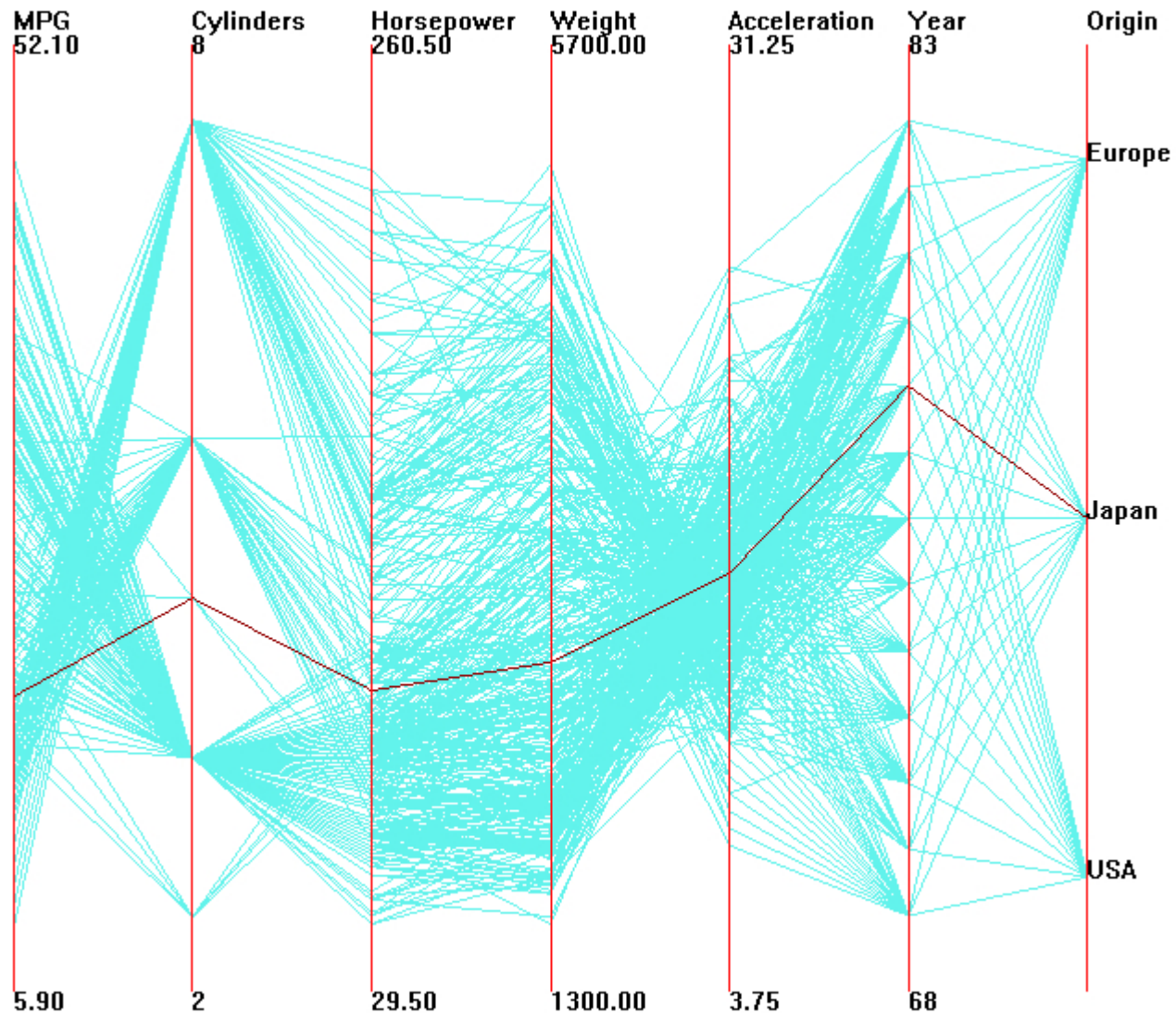
Line Charts

- If the dimensions have the same units, it is possible to use one of the previous techniques
- However, if the individual variables have different units, it is necessary to use different approach, e.g.:
 - Using multiple vertical axes
 - Vertical stacking of charts for individual dimensions

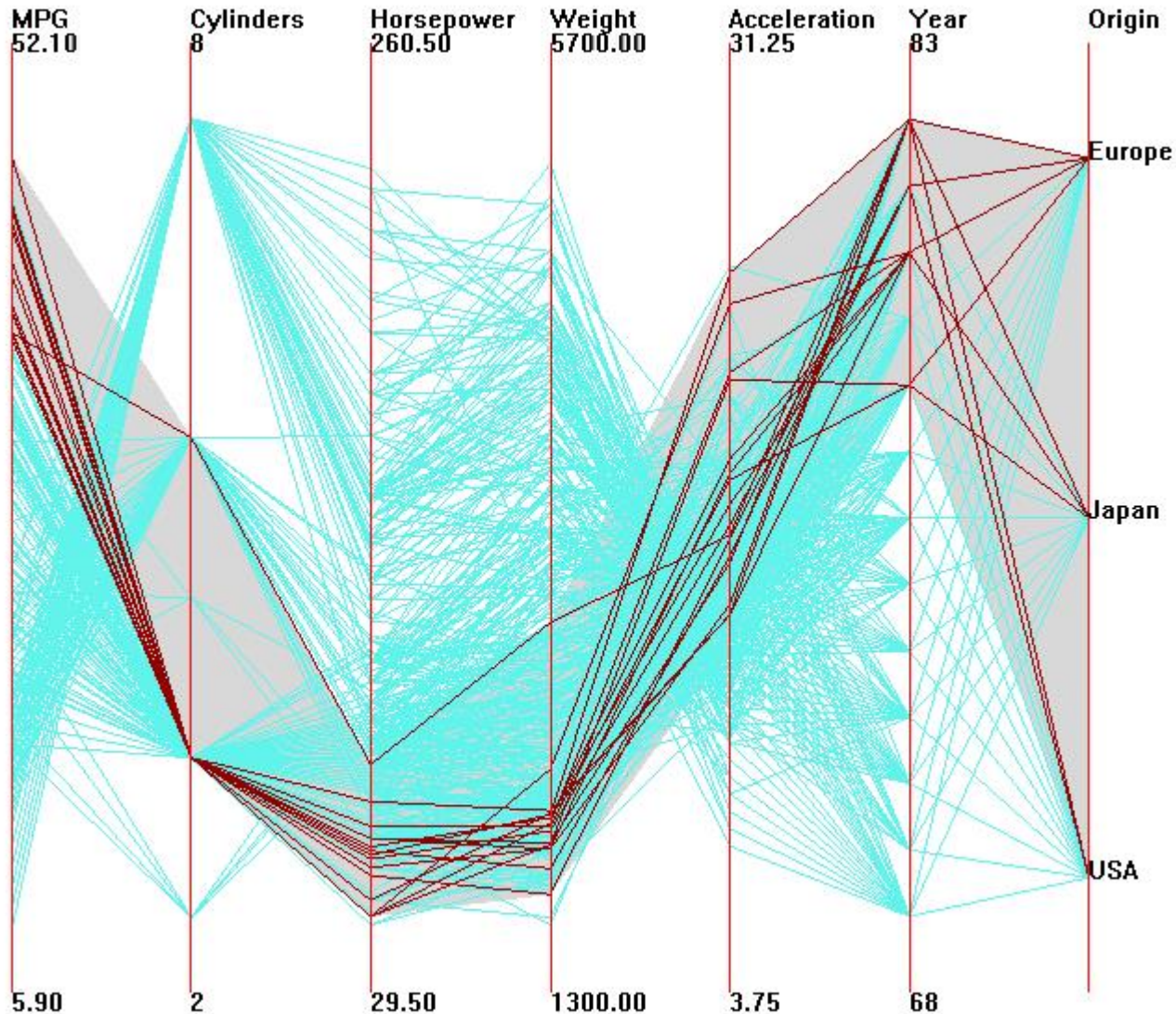
Parallel Coordinates

- Introduced in 1985 (Inselberg) as a mechanism for studying the geometry of higher dimensions
- Extending methods for analysis of multivariate data
- Instead of orthogonal placement, axes are placed parallelly next to each other
- Data record is depicted as a polyline, which crosses each axis at the position corresponding to its value in the given dimension

Parallel Coordinates

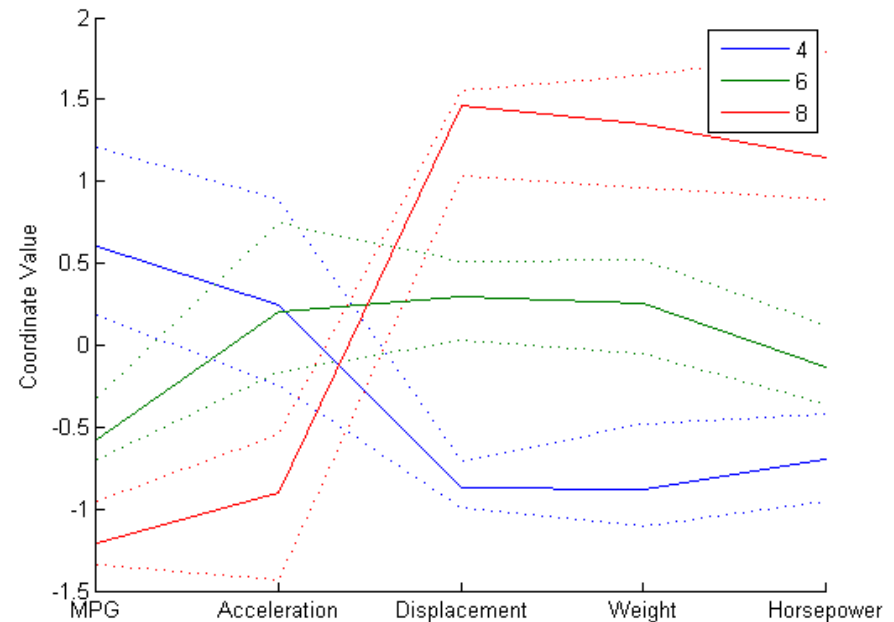
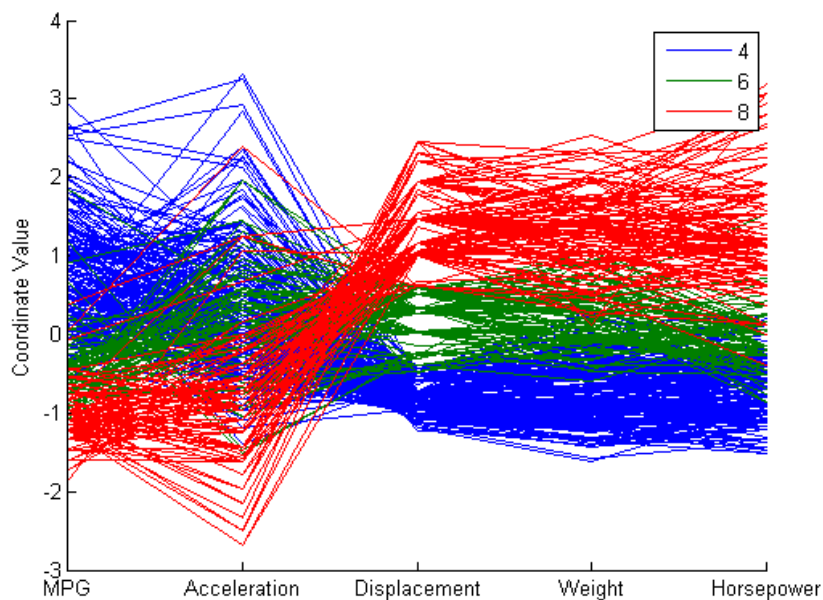


Parallel Coordinates – Interactive Selection



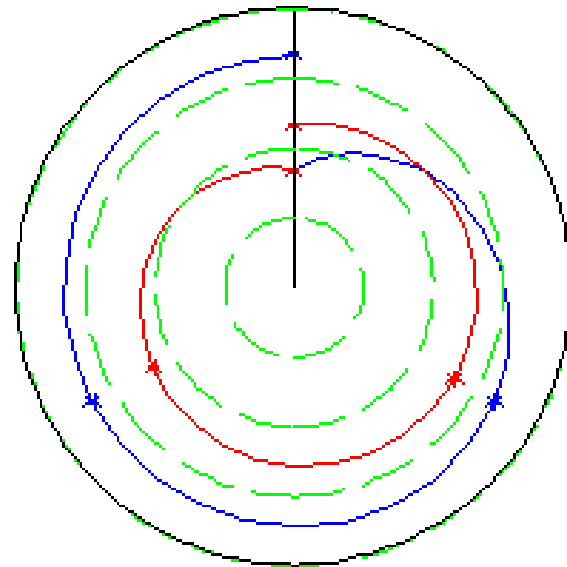
Parallel Coordinates – Median

- Become too cluttered with large amount of data



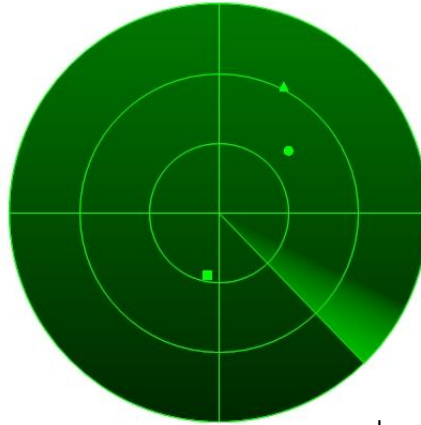
Radial Axis Techniques

- For each technique with horizontal and/or vertical orientation of coordinate system there exists equivalent technique using radial orientation
- Radial line chart

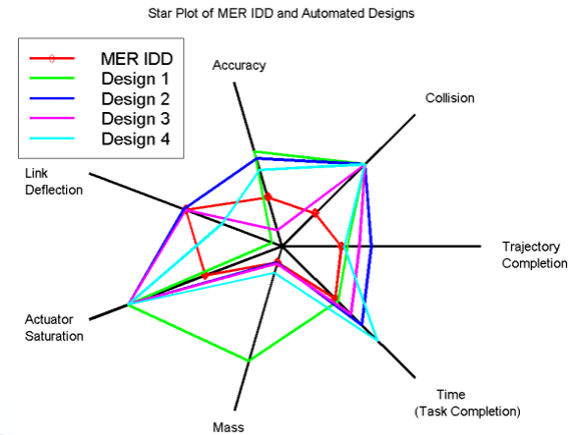


Radial Techniques

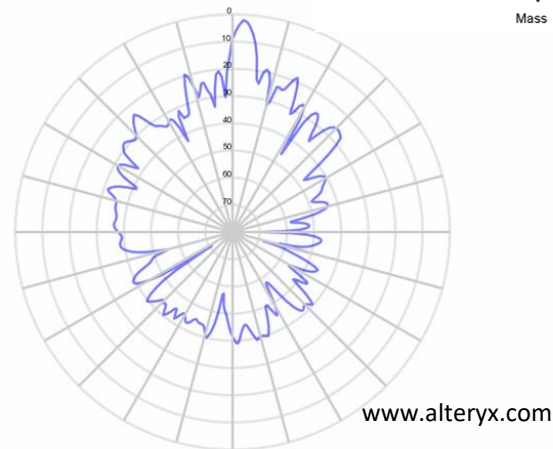
- Radar



- Star chart

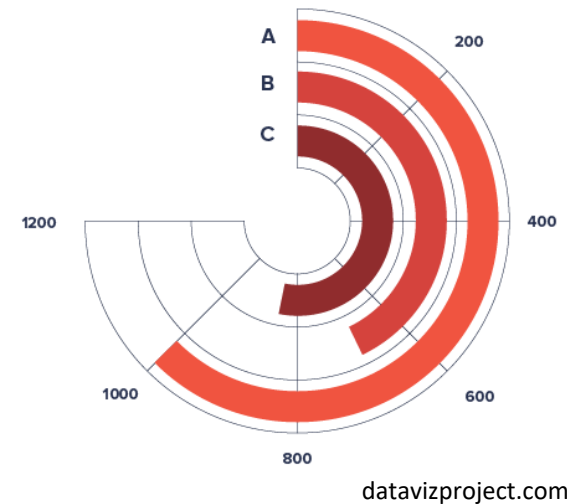
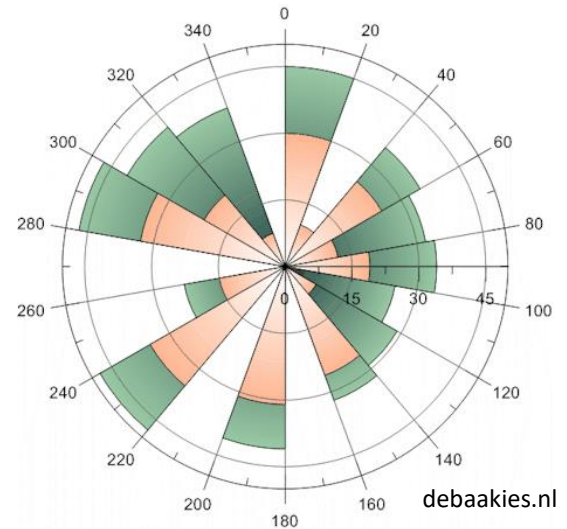


- Polar chart
 - Displaying polar coordinates



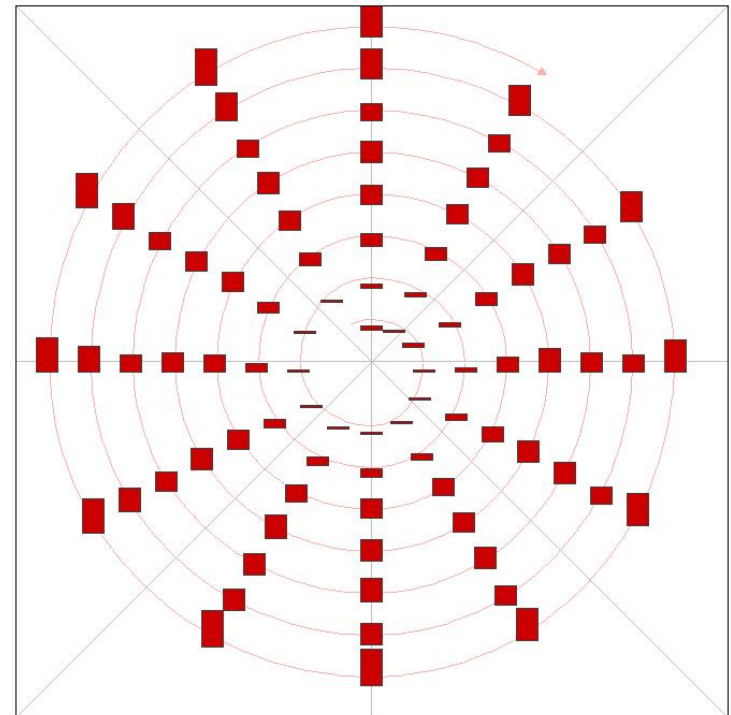
Radial Techniques

- Radial column charts
- Radial bar charts
- Radial area charts



Types of Techniques for Radial Axes

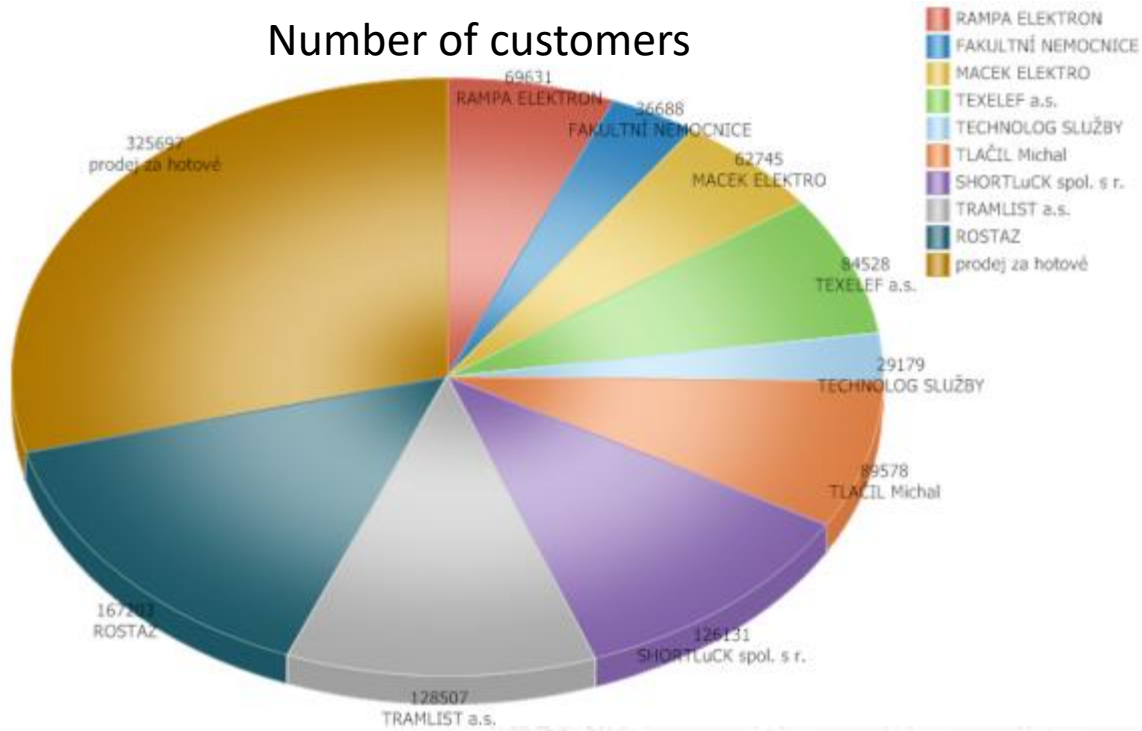
- Concentric circles
- Continuous spiral – does not exhibit discontinuity at the end of each cycle
- Compared to traditional bar representation enables observation of patterns between elements at the same position in different cycles



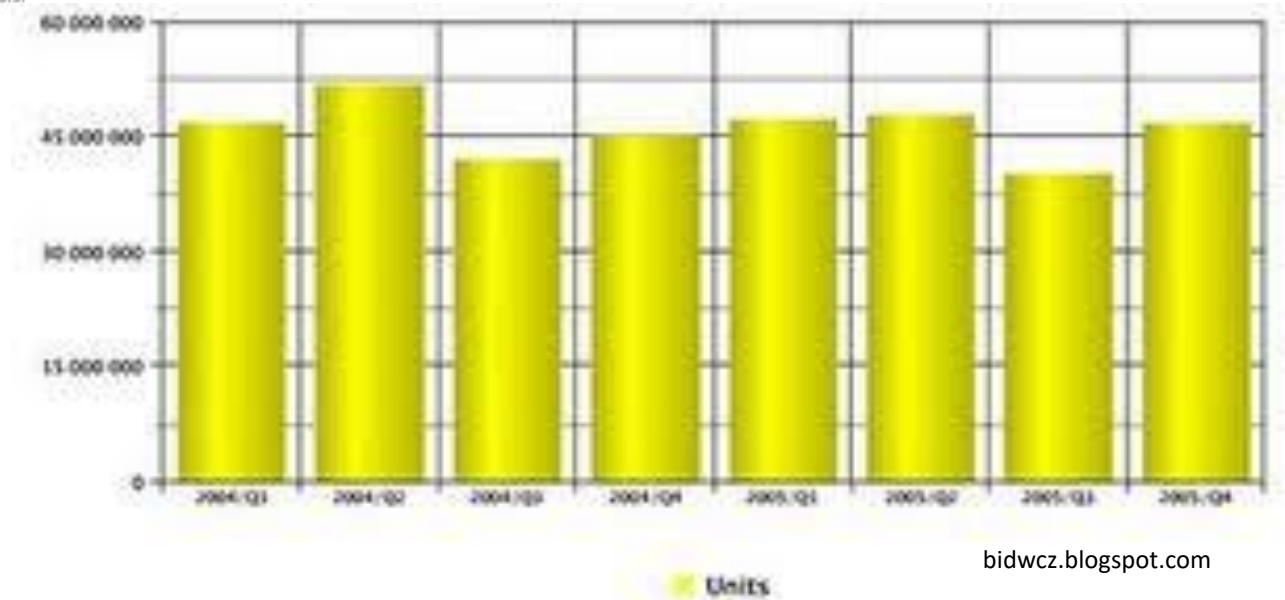
Techniques for Area Data

- Usage of filled polygons of given size, shape, color, ...
- The aim of some of these techniques is not showing individual data records, but their clusters and distribution
- Original designed for univariate data (single variable) – pie charts and bar charts. Subsequently extended for multiple dimensions.

Number of customers



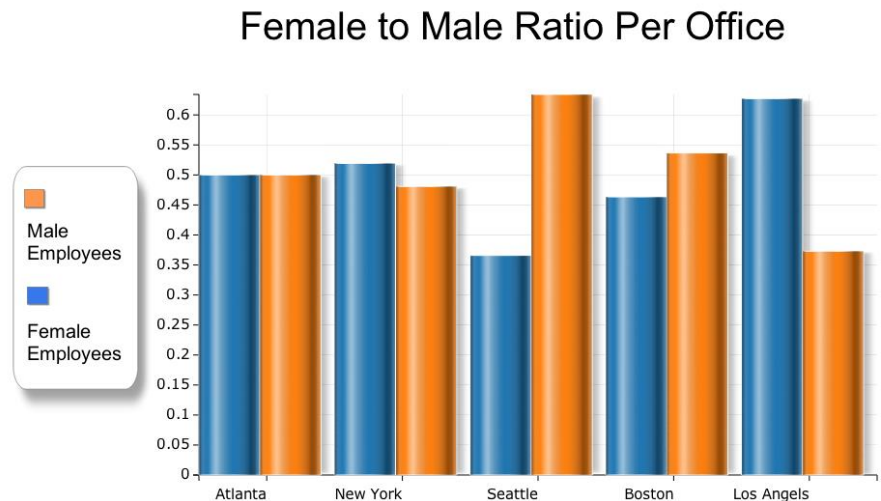
red.helios.eu



bidwcz.blogspot.com

Bar Charts/Histograms

- Rectangular columns used for displaying numerical values
- Effective thanks to human perception ability to distinguish the length and general linear properties well
- Textual labels are assigned to describe the bars

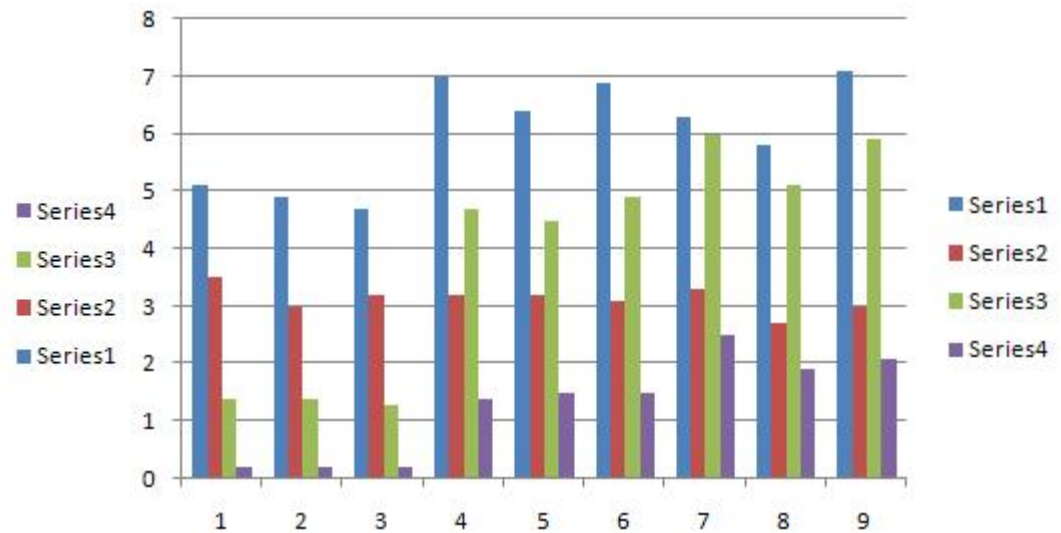
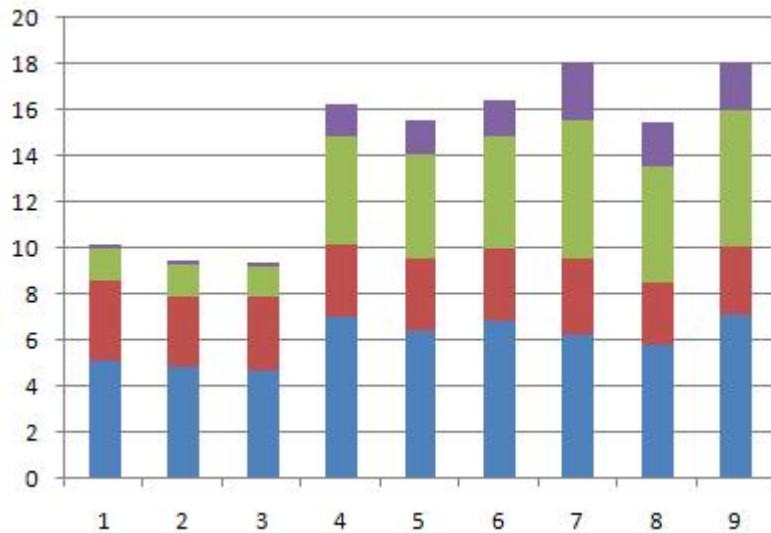


Bar Charts/Histograms

- Determining the number of necessary bars for the best data representation is essential
- Given N variables, if N is not too big, we can use 1:1 mapping
- For displaying summary or distribution of dataset we can use **histogram**
- Nominal values – the number of bars is equal to the number of different values
- Ordinal values – creating intervals of values, each interval corresponds to one bar

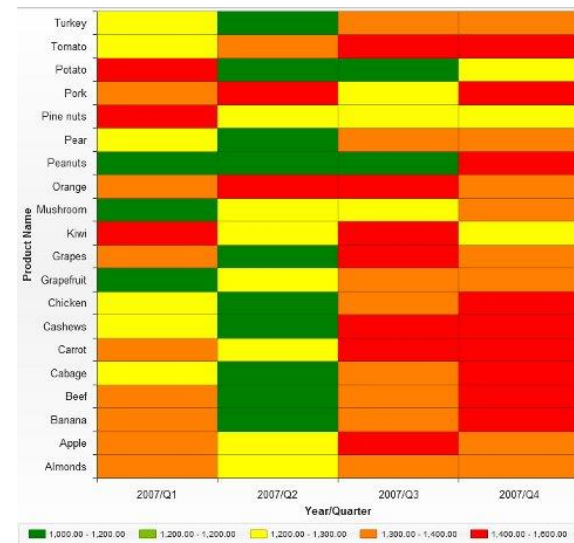
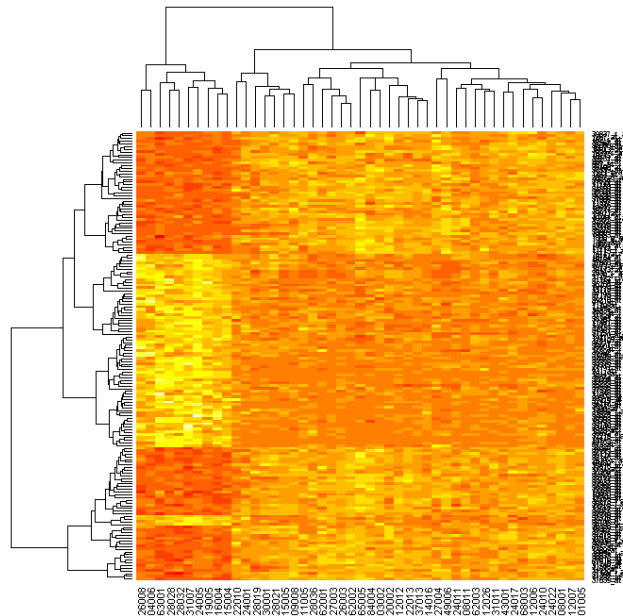
Bar Charts/Histograms

- Multivariate data – stacked bar chart

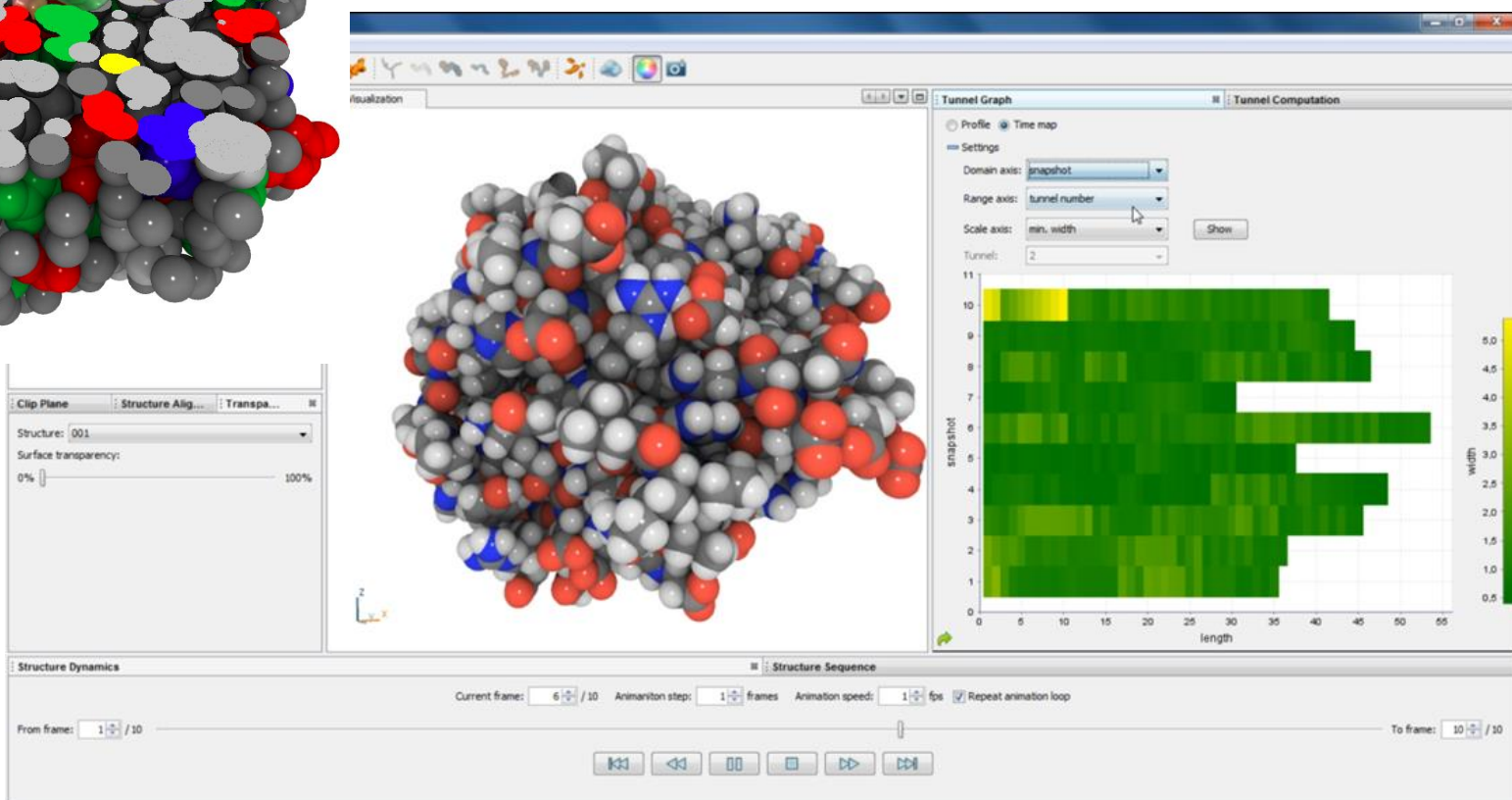
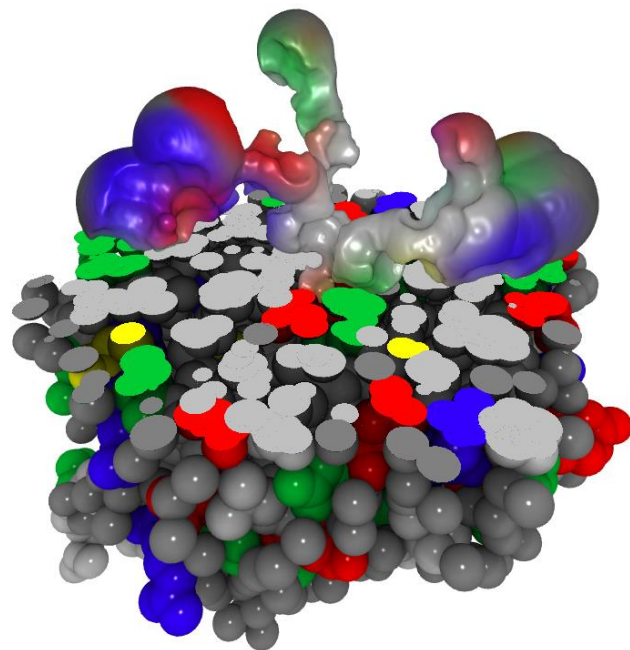


Tabular Visualizations

- Multivariate data often in tables
- **Heatmaps**
 - displaying records using color instead of text
 - each value is rendered as a colored rectangle

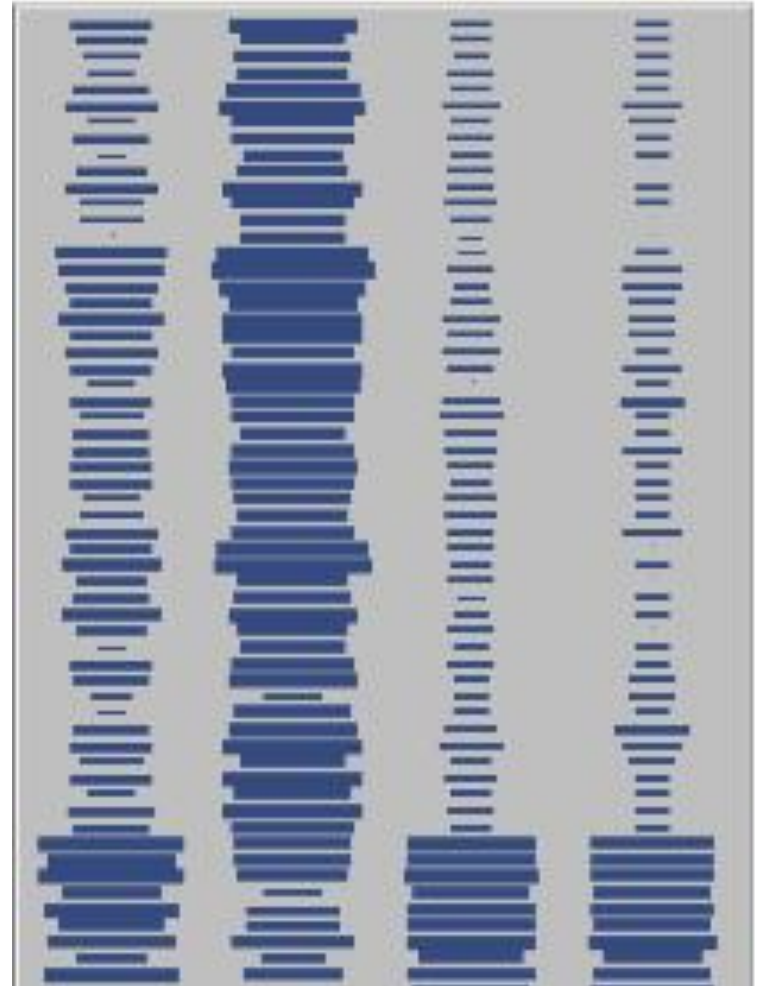


Example of Application



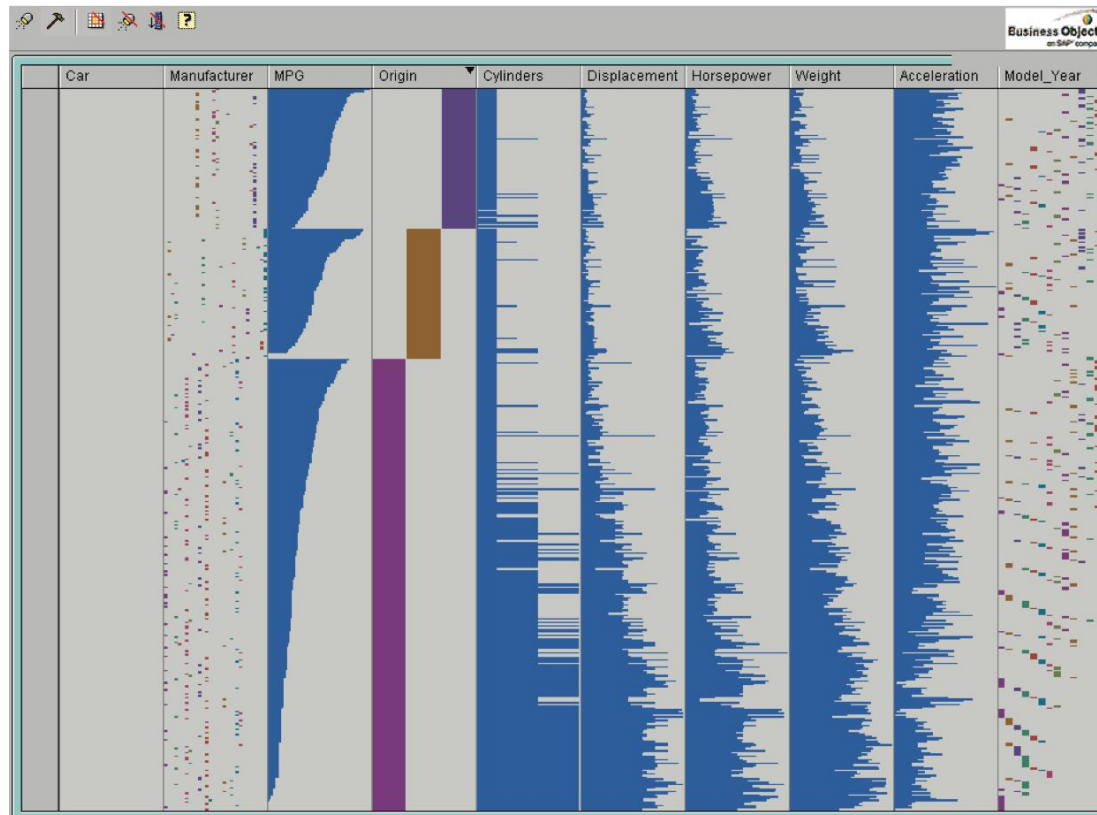
Tabular Visualizations

- **Survey plot**
 - Instead of color, the size of the cell depicts the value
 - Centres of the cells are aligned to individual attributes
 - Measurement of area is more prone to errors than measurement of length



Tabular Visualizations

- Combination of aforementioned methods into level-of-detail technique



Dimensional Stacking

- Mapping of data from discrete N-dimensional space to 2D image in such way, that the data occlusions are minimalized, while the majority of the spatial information is preserved

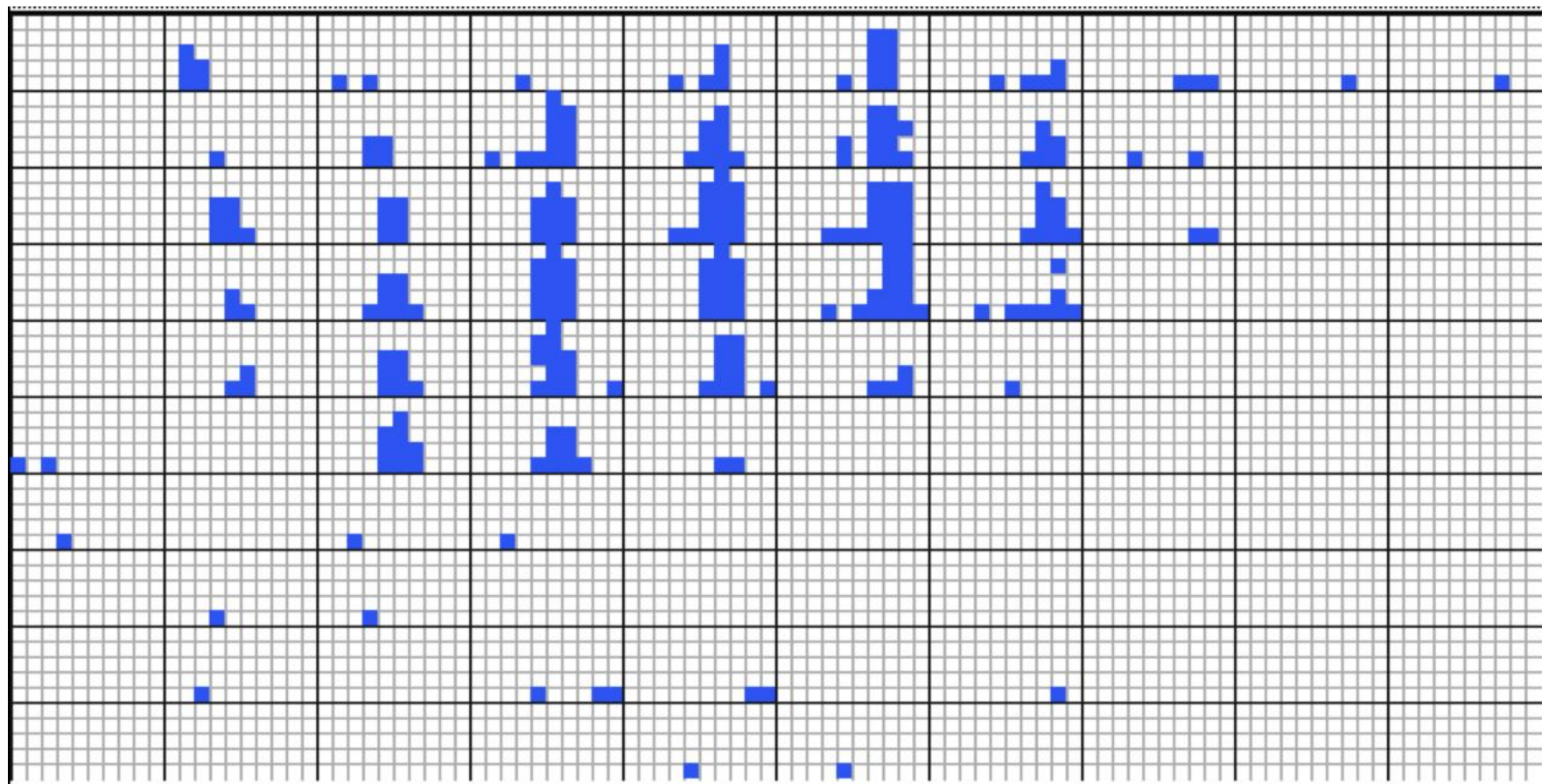
Dimensional Stacking

- Data of $2N+1$ dimensions
- Select final cardinality for each dimension
- Select one dimension as dependent variable, the rest of the dimensions are independent
- Create ordered pairs of independent variables (N pairs) and assign unique value (speed) to each pair – from 1 to N
- Pair corresponding to speed 1 creates virtual image with size corresponding to the cardinality of its dimensions
- In each position of this virtual image, new virtual image corresponding to the dimensions of pair with the speed 2 is created
- The process is repeated, until all dimensions are not included

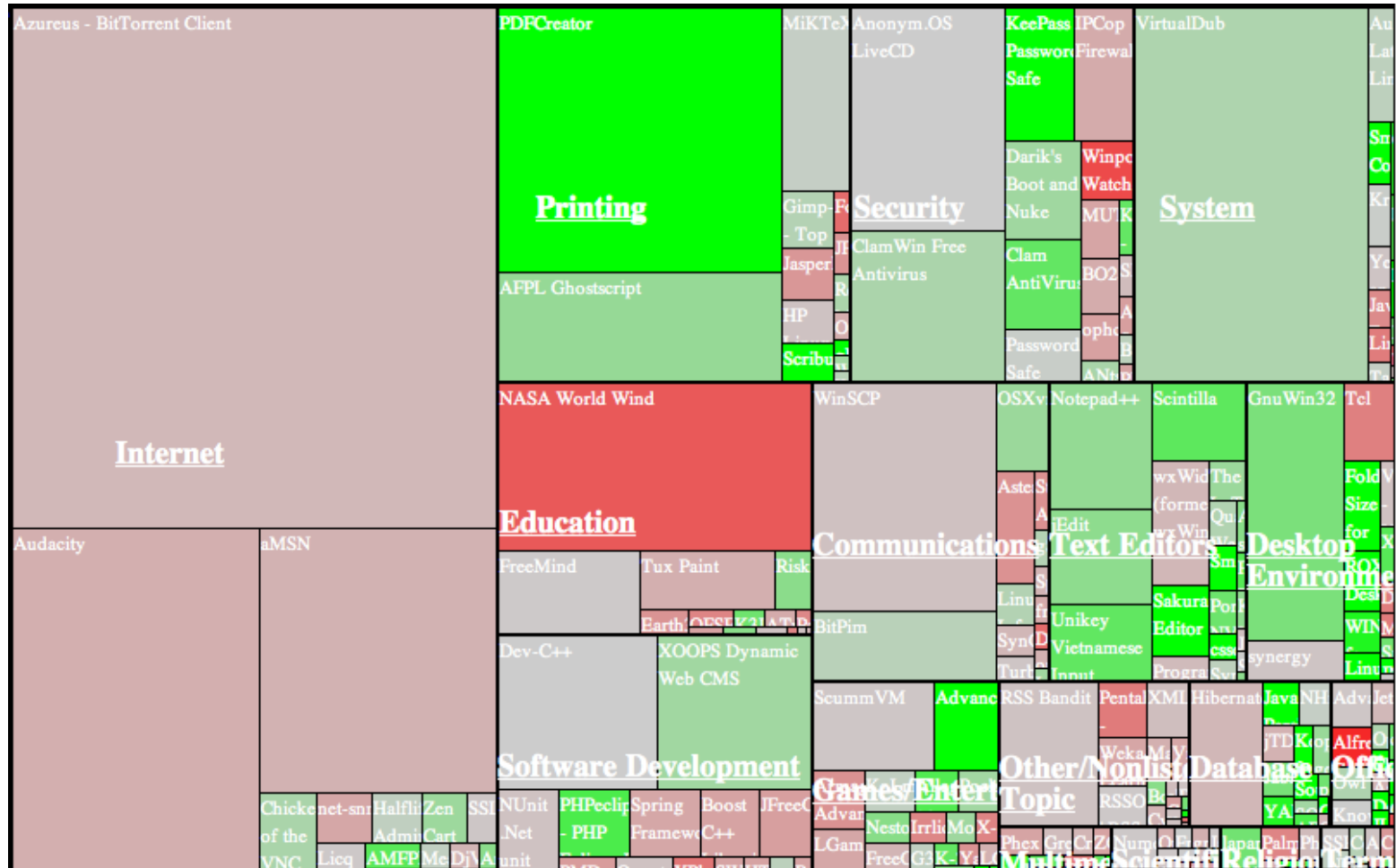
Dimensional Stacking

- Begins with discretisation of the range of each dimension. Orientation and order is then assigned to each dimension. Dimensions with two lowest orders are then used to split the virtual screen into sections - the cardinality of the dimensions indicates, how many sections are generated on horizontal and vertical axes. Each generated section is then used for recursive splitting of virtual screen in next two dimensions in the same way. This process is repeated until all the dimensions are not processed and the data are not placed to their corresponding positions on the screen.

Dimensional Stacking



Treemap

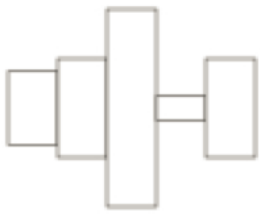
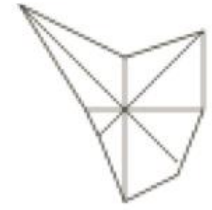
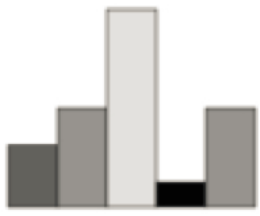
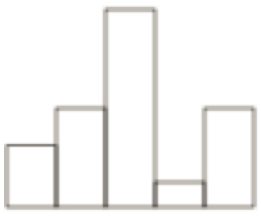


Glyphs and Icons

- Visual representation of parts of data or information, where graphical entity and its attributes are driven by one or more attributes of input data
- Graphical attributes, to which the data values can be mapped:
 - position, size, shape, orientation, material, line style, dynamics

Glyphs and Icons

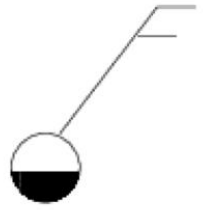
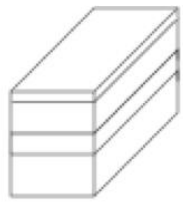
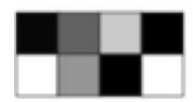
- Types of mapping:
 - 1:1 – each data attribute is mapped to unique graphical attribute
 - 1:N – set of redundant mappings (e.g., mapping data attribute simultaneously to size and color)
 - M:N – multiple or all data attributes mapped to a common type of graphical attribute



PROFILE GLYPHS

**STARS AND
METROGLYPHS**

STICKS AND TREES



AUTOGLYPH/BOX GLYPH

FACE GLYPHS

ARROWS/WEATHERVANES

Glyphs and Icons

- We must be aware of inaccuracies and restrictions of these techniques:
 - Inaccuracy of perception – depends on the type of used graphical attributes
 - Distance between graphical attributes influences the accuracy of their comparison – the closer, the more precise comparison
 - Number of dimensions and data records which can be effectively displayed using glyphs is limited

Glyphs and Icons

- After selection of the type of glyph there are $N!$ possible orderings of the dimensions, which can be used when mapping
- Several strategies for selection of suitable order exist:
 - Sorting of dimensions based on their correlation
 - Increasing influence of glyph with symmetrical shape
 - Sorting by the values of dimensions in a single record
 - Manual sorting based on knowledge of the domain

Placement of Glyphs

- Three basic types of strategies for placement of glyphs on the screen:
 1. Uniform
 2. Data-driven
 3. Structure-driven

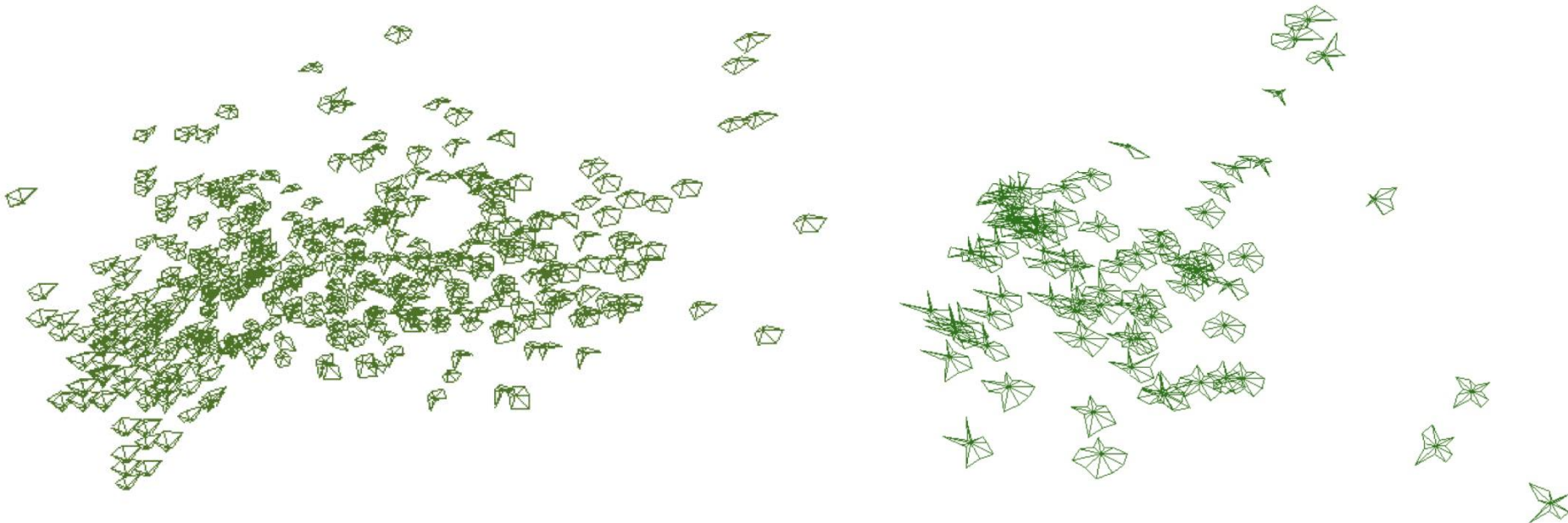
Uniform Placement

- Uniform placement on screen
- Elimination of overlaps, effective usage of screen space



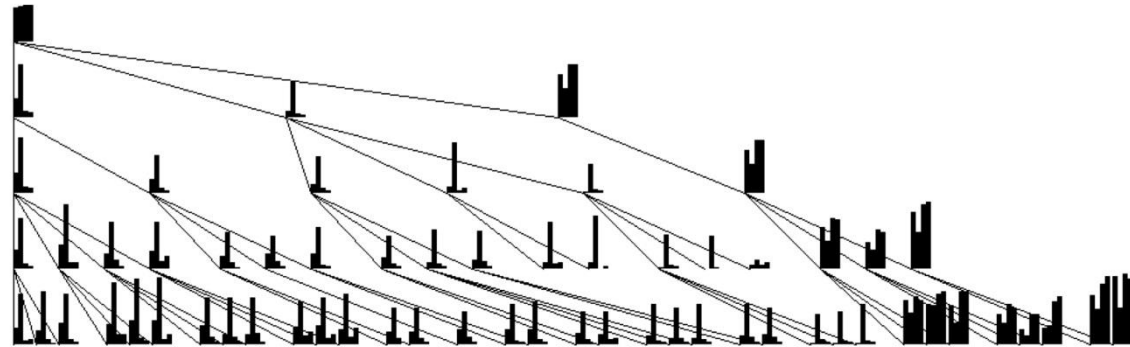
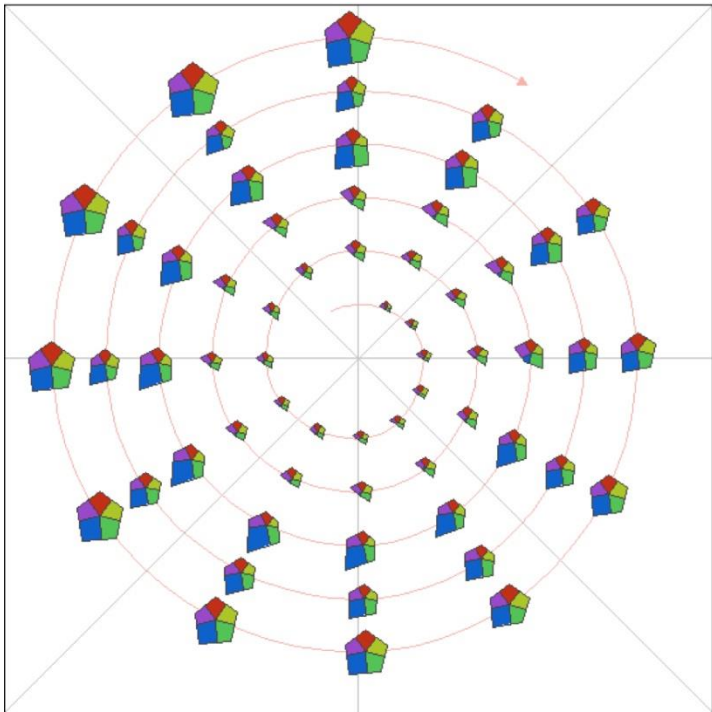
Data-Driven Placement

- Two approaches:
 - Select two dimensions to direct the placement
 - Positions derived using PCA, MDS



Structure-Driven Placement

- Using structure of the data – cyclic, hierarchical



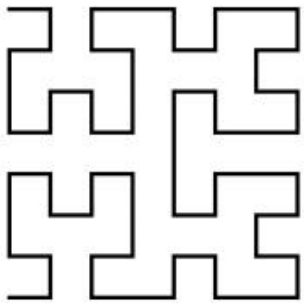
Dense Pixel Displays

- Hybrid method between point-based and regional (area-based) methods
- Maps each value to individual pixel and for each dimension creates filled polygon
- Displaying millions of values within one screen
- Number of data points determines the number of individual items in the image
- The technique relies on application of color

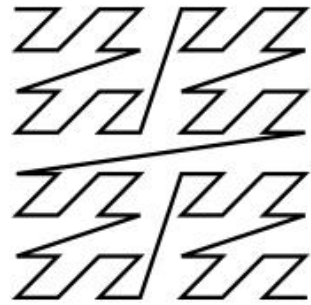
Dense Pixel Displays

- Simplest form:
 - Each dimension of dataset generates independent separated “sub-image” on the screen
 - Each dimension can be considered as an independent set of numbers, each set determines the color of the corresponding pixels
 - The placement of the items within the set (highlighting relationships between close points): alternating passes from right to left and from left to right; if the edge of the image is reached, move to the next line

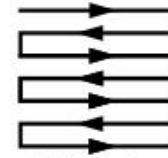
Dense Pixel Displays



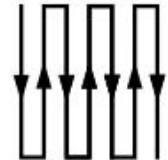
Peano-Hilbert



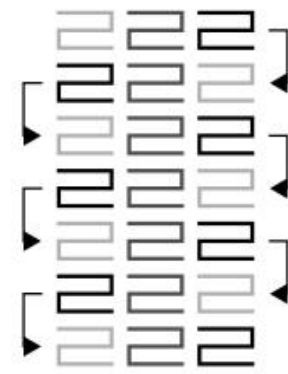
Morton



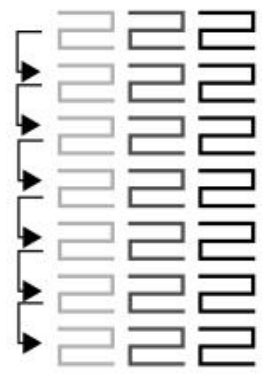
left-right



top-down



back-and-forth loop



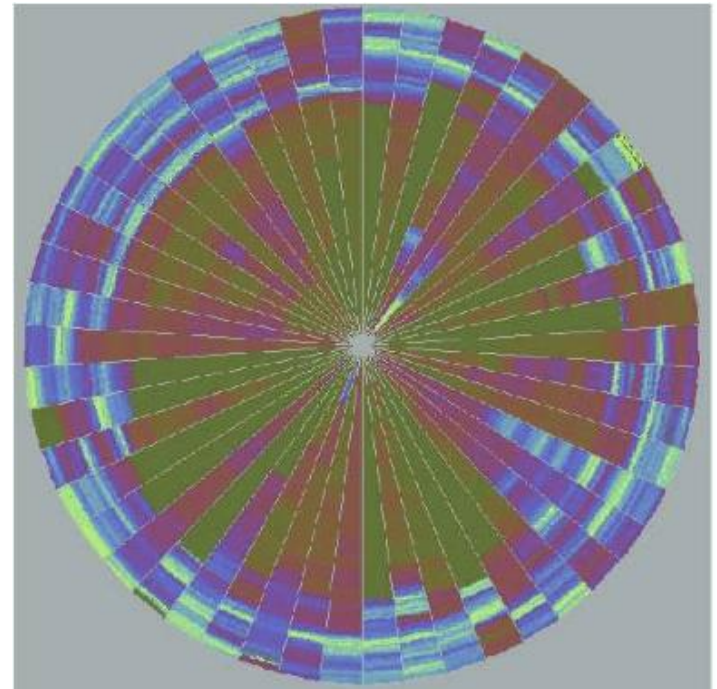
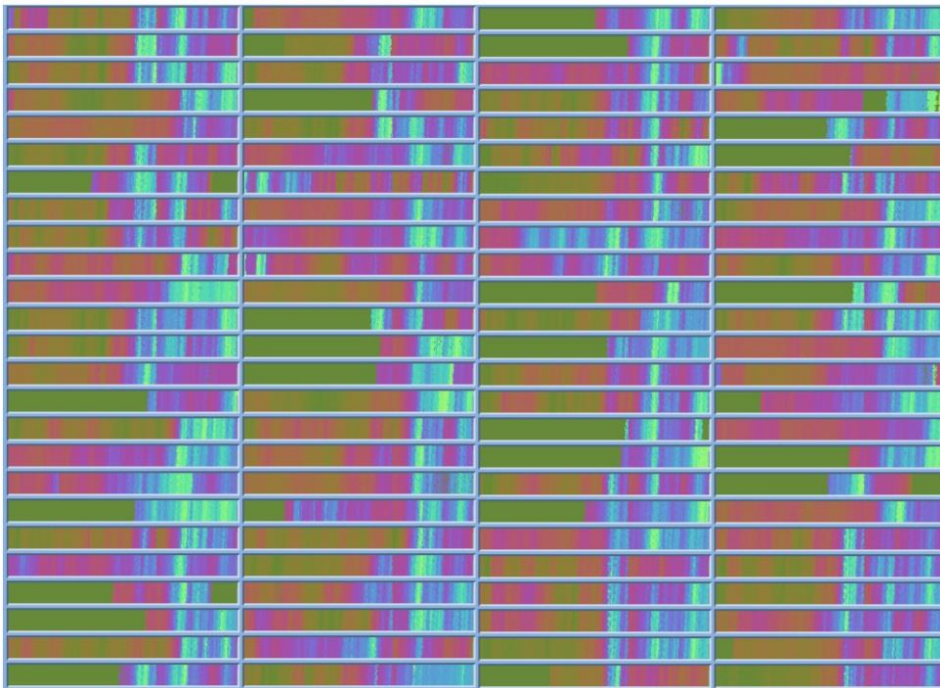
line-by-line loop

screen filling

recursive patterns

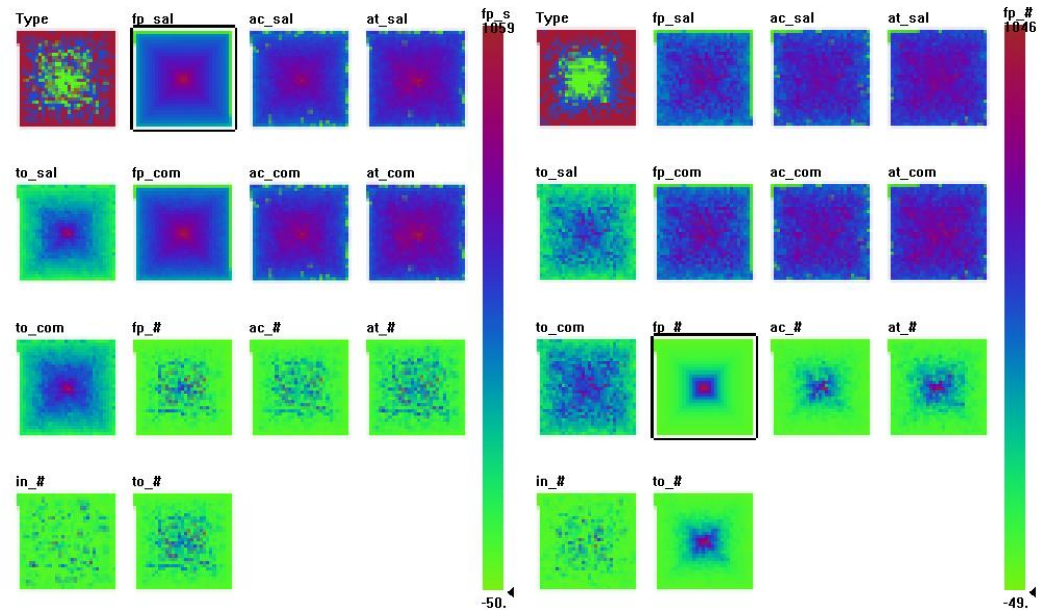
Recursive Patterns, Circular Segments

- Placement of sub-images using different approaches:



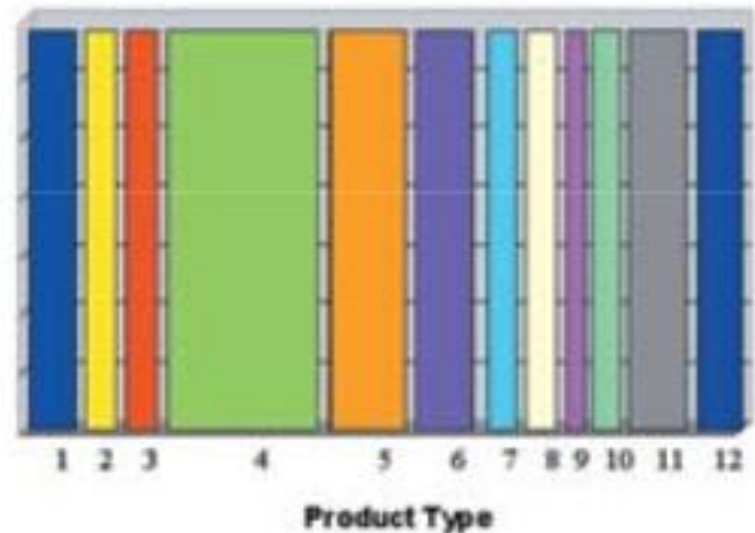
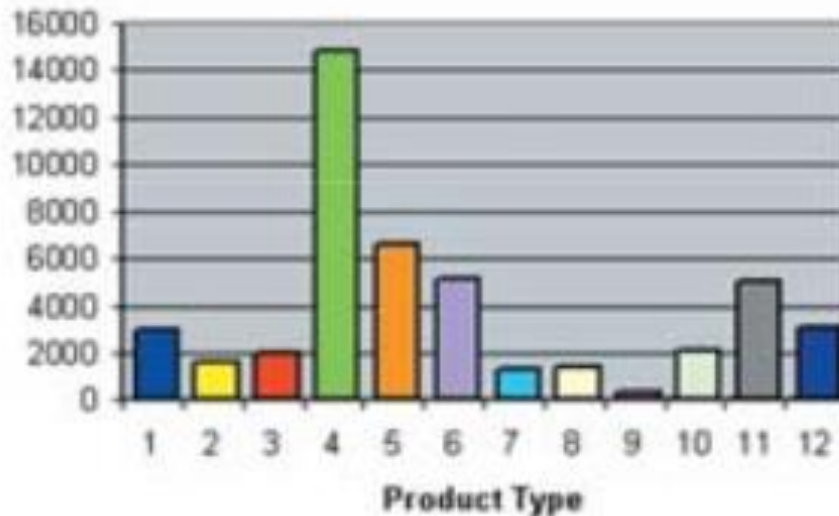
Dense Pixel Displays

- Last important aspect is ordering of the data
- Time-series data have fixed ordering
- In other types of data the change of order can reveal interesting properties



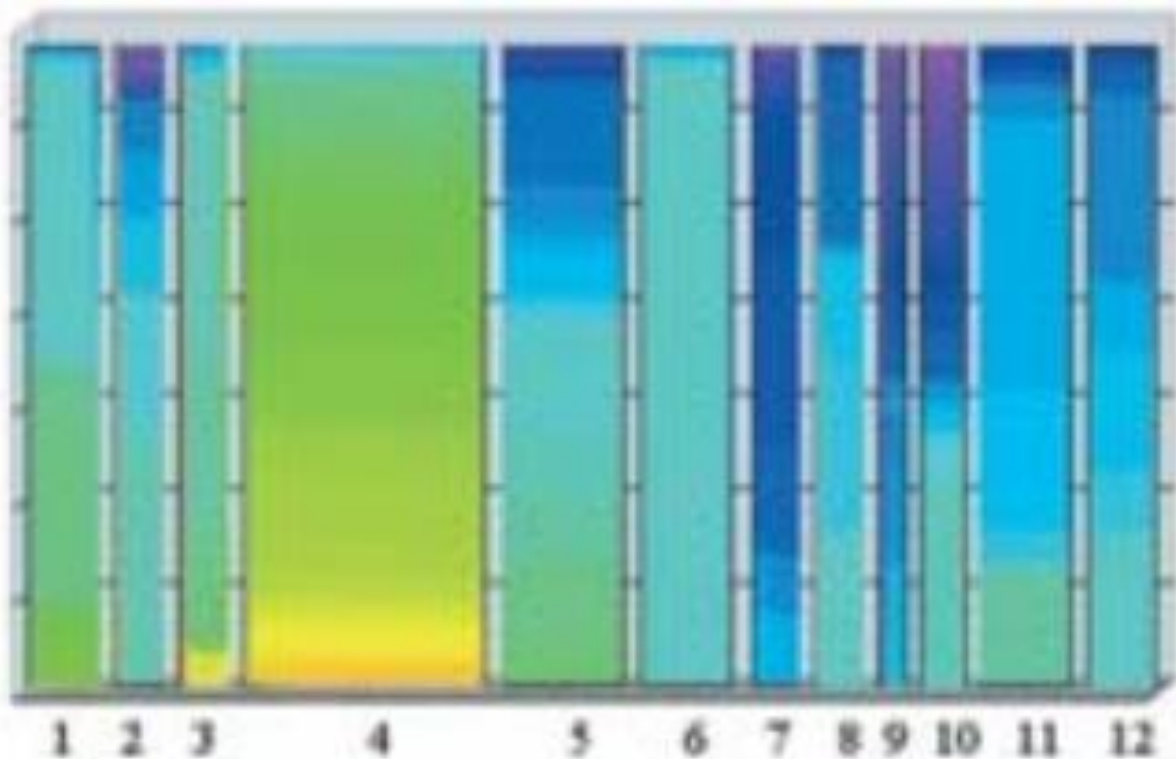
Pixel Bar Charts

- Overloading of classical bar chart – including more information about individual items



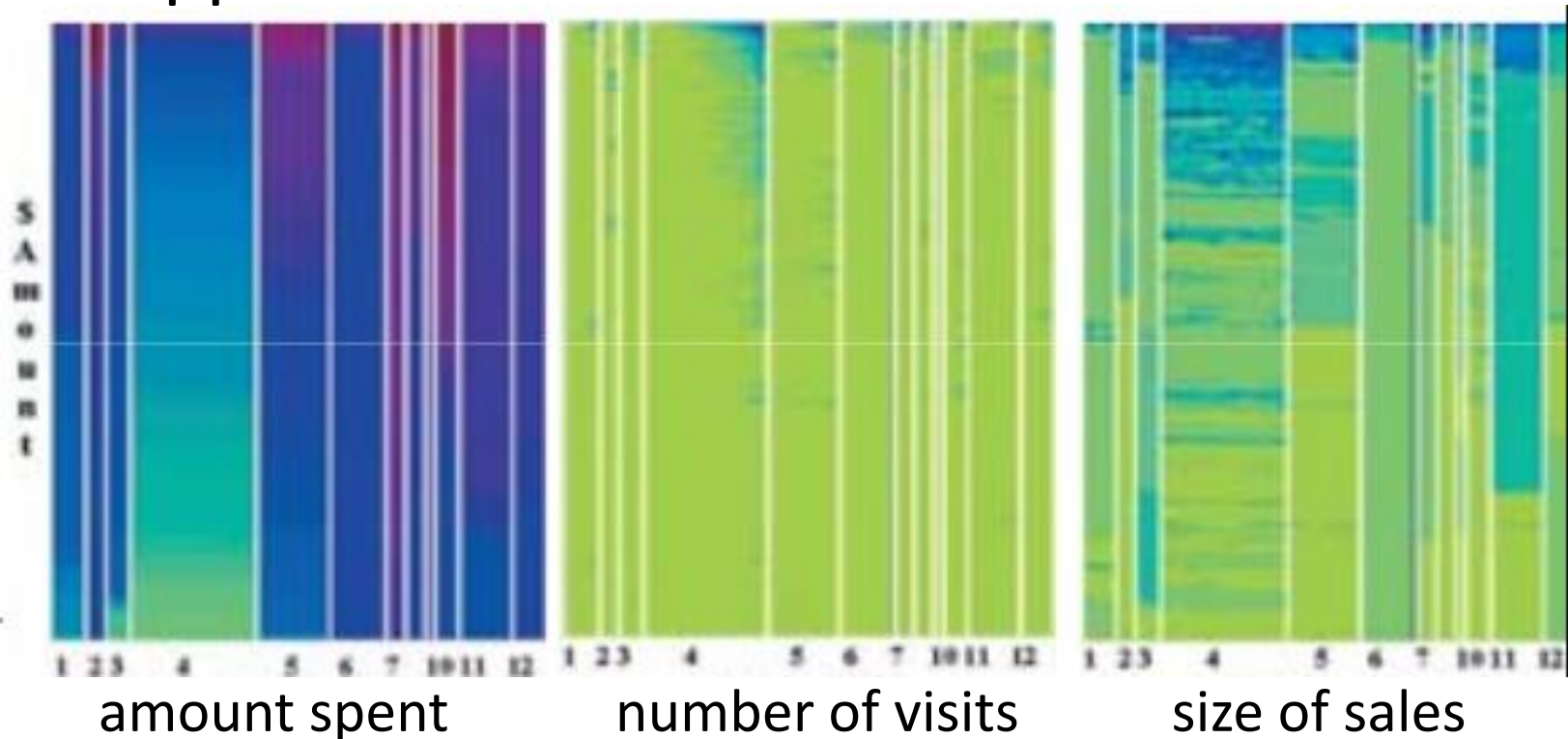
Pixel Bar Charts

- Each pixel of the bar represents a data point belonging to the group represented by this bar



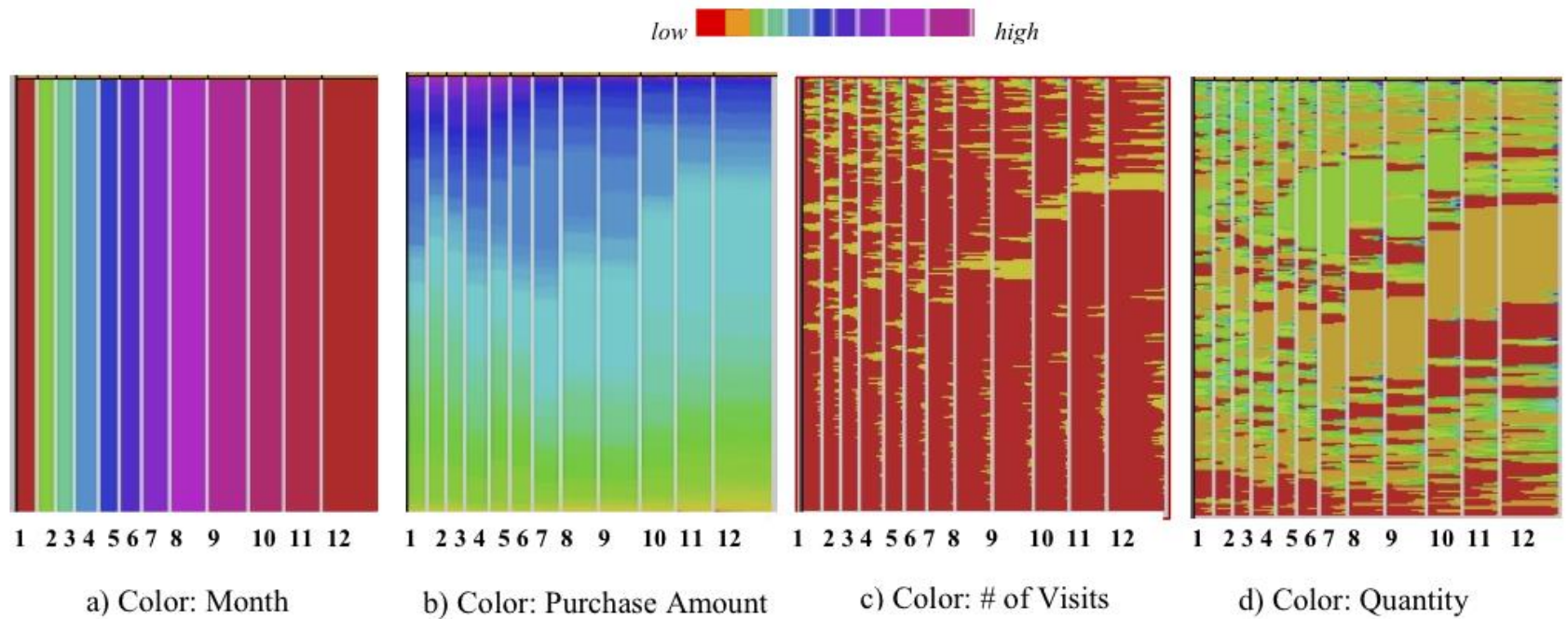
Pixel Bar Charts

- Internet shopping – relationship between the type of product and the price. Color is mapped onto:



Pixel Bar Charts

- Placement of dense pixels to bar chart



Pixel Bar Charts

- We can derive, e.g.:
 - The largest amount of customers came in December, while in February, March, and May there was minimum of customers.
 - From February to May there were largest amounts of purchases.
 - Number of purchases in December is average.
 - From march to June the customers returned more frequently than in other moths. December customers were mostly one-time customers.
 - Customers shopping the most are returning more often and buying more stuff.