Data Visualization: Ideas, Methods, and Problems

Vydunas ŠALTENIS,
Vilnius Pedagogical University
Sindoos 39, LT-2600 Vilnius, Lithuania
e-mail: saltenis@kit.mii.lt

Jūratė AUŠRAITĖ
Institute of Mathematics and Informatics
Akademijos 4, LT-25221 Vilnius, Lithuania
e-mail: jurotae@vmsl.lt

Received: July 2002

Abstract. The science of data visualization requires both a theory of perception and of computer graphics. This article explores the convergence sciences of perception and techniques of visualization. Benin’s Image Theory of visualization research offers a promising technology for transforming an indigestible mass of numbers into a medium which human beings can understand, interpret, and explore.

Visualization techniques are of increasing importance in exploring and analyzing large amounts of multidimensional information. In the article we try to set some classes of visualization techniques. We describe some of the most popular multidimensional data research methods, and present some examples.

Visualization is applied in education of old. It must make connections between knowledge the learner has and the knowledge being taught. Therefore in order to design effective visualizations it is necessary to know (or at least have a theory about) what the learner knows. This is especially important in the context of education.

Key words: data visualization, proximity visualization, clustering methods, projection methods, multidimensional scaling, self-organizing map.

1. Introduction

Visualization, a term used in the industry since the publication of the National Science Foundation report Visualization in Scientific Computing (McCormick et al., 1987), is a form of communication that transcends application and technological boundaries. With the advent of computer graphics, researchers can convert entire fields of variables (representing density, pressure, velocity, entropy and so on) to color images. The information conveyed to the researcher undergoes a qualitative change because it brings the eye-brain system, with its great pattern-recognition capabilities, into play in a way that is impossible with numerical data alone. Data visualization is an established technique for exploration, analysis, and presentation of data.
For example, an observer instantly sees the vortices, shock systems, and flow patterns in the visualization of hydrodynamic calculation, while these same patterns are invisible in mere listings of several hundred thousand numbers, each representing field quantities at one moment of a time. When computing the space-time solution of the laws of physics, particular numerical quantities at each event in time-space are not important; rather, what is important is understanding of the global structure of the field variables that constitute the solution and the causal interconnections of the various components of it.

At present, visualization is used in various scientific and engineering research areas (DeFanti et al., 1989), for example:
- the use of interactive computer graphics in chemistry, which began in 1964;
- scientific computation applied to medical imaging has created opportunities in diagnostic medicine;
- visualization helps mathematicians to understand various equations that are too complex to conceptualize otherwise;
- meteorologists obtain information on natural behavior that cannot be safely observed: differing vertical wind, temperature, pressure and moisture structures, etc.;
- the planetary study involves the accumulation of huge volumes of data on the planets in the solar system.

2. Theory of Perception in Data Visualization

The science of data visualization requires both a theory of perception and of computer graphics. However, visualization designers have paid relatively little attention to perceptual issues. Bertin’s (1983) Image Theory can serve as a guide to visualization design. Visualization research offers a promising technology for transforming an indigestible mass of numbers into a medium which humans can understand, interpret, and explore.

As shown in Fig. 1, the first stage maps numbers (data/process) to images by means of some algorithmic technique. The second stage maps images to insight by means of perception.

```
<table>
<thead>
<tr>
<th>Data/Process</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer/Algorithms</td>
</tr>
<tr>
<td>Graphics/Imaging</td>
</tr>
<tr>
<td>Psychology/Perception</td>
</tr>
<tr>
<td>Insight</td>
</tr>
</tbody>
</table>
```

Fig. 1. The transformation from numbers to insight requires two stages.

2.1. Bertin’s Image Theory Concepts

Bertin’s key concept is image, from which the theory derives its name. An image is the fundamental perceptual unit of visualization. In terms of Bertin, three stages of information are extracted (see Fig. 2):

1) external identification is that the graph is showing something, for example, height and weight;
2) internal identification is perception that the height component is mapped onto the vertical axis of the plane and weight onto the horizontal axis; and
3) perception of correspondence is noting that the data point is located at a particular intersection of visual variables X and Y.

According to Bertin, there are two functionally different classes of visual variables: planar and retinal. Components may be represented by six retinal variables: size, color, shape, orientation, texture, and brightness. Each retinal variable can be used with three types of implantation. Bertin believes that an efficient visualization is limited to three-component visualizations. It is not possible to create an efficient four-component visualization by adding a second retinal variable: you could not, for example, add a component for country of birth by coding each mark with the shape reflecting nationality.

Bertin believes that the major problem of most data visualizations is the choice of visual variables with an inappropriate length or level of organization. Length refers to the number of categories or steps (distinguishing different colors, brightness levels, etc.) A visual variable must have a length equal to or greater than the component it represents. A major problem for the visualization designer is to match the data to a visual variable with the correct length.

The level of perceptual organization specifies the type of data scale: nominal, order, or ratio, which each visual variable can portray (Table 1). That is, suppose the goal is to allow the viewer to extract ratios from the visualization, e.g., to immediately see that one value is twice as much as another. Brightness could not be used, for example, because doubled intensity produces only about a 1.4 factor increase in the perceived brightness.

![Fig. 2. Height/Weight graph, with additional components represented by marks of different shape (gender) and brightness (nationality).](image-url)
### Table 1: Bertin Levels of Perceptual Organization

<table>
<thead>
<tr>
<th>Associative</th>
<th>Selective</th>
<th>Ordered</th>
<th>Quantitative</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planar</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Size</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Brightness</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Texture</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Color</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Orientation</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Shape</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Associative Organization.** Bertin subdivides the nominal scale level into two subcategories: associative and selective. Associative perception is the lowest organization level, allowing grouping of all the elements of a variable in spite of different values. According to Bertin, planar dimensions, texture, color, orientation (point and line implants only), and shape are associative, while size and brightness are not.

**Selective Organization.** The concepts of associative and selective organization have implications for a wide range of issues in data visualization, such as the presentation of multiple data views.

Fig. 3 shows a typical example. One window shows data in graph form, while the other shows a table. It is obvious if the two views are linked by color. If the informational view in the right window had a different color than its corresponding object in the left view, the viewer would incorrectly form a nominal scale grouping.

**Ordered Organization.** The associative and selective perception provides only a nominal scale classification. Bertin suggests that additional variables allow the data to be ranked. Observers can see that one value of a variable represents a larger or a smaller quantity than another. For example, the increased age could be represented by data points with increased brightness. In this case, it would be possible to make decisions about relative height/weight of elder or younger people. Bertin considers the planar variables: brightness, size, and texture variables as ordered while excluding shape, color, and orientation.

**Quantitative Organization.** The highest level of organization is quantitative which permits a direct extraction of ratios. That is, the ratio of variable values maps directly to the ratio of the data values. If age were represented by data points of different size, which is a quantitative variable, the viewer could immediately see which individuals were twice as old as others. While ordered data permit only a relative magnitude of the represented quantities, quantitative variables support the perception of ratios. Only size and the planar dimensions are quantitative.

#### 2.2. M. Green’s Extension Beyond the Bertin Image Theory

In his work, Green (1998) tries to show that Bertin bases his theory solely on his introspection and makes no attempt to provide any empirical data support. He outlines some basic researches on the human vision and suggests the ways of using them to guide data visualization.

M. Green states that:
1) advent of computer graphics permits the use of several new visual variables that Bertin does not discuss;
2) there are several secondary perceptual effects which complicate a direct application of Image theory; and
3) perhaps the most important, Bertin’s belief that images are always limited to three components is false.

In Table 2, M. Green uses research data to propose corrections and extensions to Bertin’s original level of organization assignments (Table 1).

**Shape can be selective.** The results of research (Green, 1991) show that the list of primitives largely reflects those coded by the brain in the primary visual cortex. Accordingly, the list agrees well, but not perfectly, with Bertin’s list of retinal variables. The image is typically processed by a series of cascaded linear and nonlinear filters, which

#### Table 2: M. Green’s Updated Bertin Levels of Perceptual Organization

<table>
<thead>
<tr>
<th></th>
<th>Associative</th>
<th>Selective</th>
<th>Ordered</th>
<th>Quantitative</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planar</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Size</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Brightness</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Texture</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Color (Hue)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes-limited</td>
<td></td>
</tr>
<tr>
<td>Orientation</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shape</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Motion: Velocity</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes-limited</td>
<td></td>
</tr>
<tr>
<td>Motion: Direction</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Flicker: Frequency</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes-limited</td>
<td></td>
</tr>
<tr>
<td>Flicker: Phase</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Disparity</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 3. One window shows data in graph form, while the other shows a table.
presumably correspond to neuronal levels in the brain. The decision as to whether two shapes/textures are associative or selective is determined by the similarity of their outputs in the model. In theory, such a model should make it possible to predict which shapes/textures would be associative or selective. The applicability of such models to predicting immediate visualization of diagrams is unknown.

**Color can be ordered.** Color consists of three properties: hue, saturation, and brightness (or hue, chroma, and lightness). It is true that hue is in general a nominal variable: red, green, blue do not form an ordered scale. But, over small ranges, hue can be ordered. For each hue, there is a specific example, called a “unique hue”, which is perfect in that it is not tinged by any other hue. For example, unique yellow contains no trace of either red or green. An ordered scale of yellow could therefore be constructed starting around unique yellow and extending to either (but not both) unique red or unique green. Secondly, saturation, the amount of white mixed with a spectral, i.e., the purest, hue, is also an ordered variable.

**Brightness can be “quantitative.”** Brightness is ordered but not quantitative because it produces a psychometric function with an exponent less than one. It should be easy, using existing psychophysical data, to promote brightness to a quantitative level of organization by rescaling the intensity axis. The visual system’s brightness exponent varies with conditions and can be as low as 0.33.

M. Green (1998) proposes that data can be displayed by means of several new visual variables, including motion, flicker, and disparity (stereoscopic depth).

**Motion.** One new retinal variable is motion, which can be split into two subvariables: velocity and direction. It is unlikely that motion is associative. Objects with different motions are generally perceived as lying on different surfaces. Segmentation of an image into constituent surfaces is perhaps the earliest and the most primitive perceptual function. Motion velocity is likely to be ordered, since it is a continuum of magnitude, and observers can readily discriminate steps of increasing value. Direction of motion is likely to be unordered, since it is not a variable of magnitude. Velocity is not quantitative because it produces a compressive psychometric function, through which it might be made quantitative by appropriate rescaling.

**Flicker.** Like motion, flicker also has two subvariables - frequency and phase. Flicker frequency, the speed of the on-off cycles, is selective by common sense, can be readily ordered, and can be promoted to quantitative by rescaling. There are some problems with flicker frequency. First, it has a very short length. Secondly, apparent brightness varies with frequency, making these somewhat “integral dimensions”. Flicker frequency is not associative.

**Binocular Disparity.** A third new variable is binocular disparity, which can be created by giving the left and right eyes slightly different views of the same visualization. Instead, disparity provides good relative depth information, revealing whether one object is closer or further away than another, and is a very powerful cue for image segmentation. Its main role is to help break scenes into meaningful surfaces, so it is likely to be poor for association but ideal for selection. Since it provides only relative depth information, it is a poor choice to represent quantitative data but good for representing ordered data.

Bertin asserts that images can contain no more than three components; however, M. Green’s studies have found that four or more component images are possible. A subset of visual variables permits the conjunction search. Observers can effortlessly search for conjunctions of shape or color with disparity (Nakayama and Silverman, 1986) or motion. These searches have four components: 2 planar, 1 Bertin retinal variable, plus disparity or motion, variables which Bertin does not include in his analysis.

### 3. Proximity Visualization

**Proximity Visualization** is the visualization of abstract data collections. Database tables with many attributes, graphs, and multimedia collections are types of data collections for which it is difficult to design useful visual representations. However, similarity between elements of such collections can be measured, and a good overview picture should respect this proximity information by positioning similar elements close to each other, while far from dissimilar ones. Such a visualization is, in effect, a topology, preserving map of the underlying data collection.

#### 3.1. Data Types

The proximity of a pair of objects from a data collection can be expressed either as their similarity, mutual agreement, or dissimilarity. A measure of dissimilarity between a pair of objects from a collection is a dissimilarity coefficient. A number of coefficients could be used with a particular data collection. Each data type could be serviced by different dissimilarity coefficients.

A quantitative variable can either be measured on an interval or a ratio measurement scale, for example temperature.

An ordinal scale is weaker than quantitative in that it also induces an ordering of objects, but does not make any statement about the magnitude of the differences. An example of an ordinal scale is a well-defined ordering $A > B > C > \ldots > F$. It is impossible to know how the differences between the pairs of grades relate, for instance, whether the difference between A and B is greater than that between B and C. However, the difference between A and C must be greater than the difference between A and B, due to the inequalities, and an effective dissimilarity coefficient for a pair of objects measured on an ordinal scale can be based on differences in their rank.

A nominal scale allows for the weakest form of measurement, as it does not enforce any ordering of objects. A number of non-overlapping categories are defined and an object can belong to any single one of these. The similarity of a pair of objects is defined in terms of their category membership; if they belong to the same category, they are considered as equivalent, otherwise, they are completely different. An example of such an exclusive categorization is a person's zodiac sign.

**Binary.** A binary scale of measurement is a special case of a nominal scale with only two categories, for example, male-female.
of the data stored in databases, however, there is no standard mapping into the Cartesian coordinate system since the data have no inherent two- or three-dimensional semantics. In general, relational databases can be seen as multidimensional data sets with the attributes of a database corresponding to the dimensions.

The visualization techniques can be divided into geometric projection techniques (Inselberg, 1985; Inselberg and Dimsdale, 1990), icon-based techniques (Pickett and Grinstein, 1988; Beddow, 1990), pixel-based techniques (Keim, 1995; Keim and Kriegel, 1994; Keim et al., 1995), hierarchical techniques (LeBlanc et al., 1990; Robertson et al., 1991; Shneiderman, 1992), graph-based techniques (Eick and Wills, 1993; Becker et al., 1995), and combinations thereof (Asimow, 1985; Altberg and Shneiderman, 1994).

Geometric projection techniques display include techniques from exploratory statistics, such as scatterplot matrices (Andrews, 1972; Cleveland, 1993). Other geometric projection techniques include projection views (Burnas and Baja, 1994; Spence et al., 1995), hyperslice (Wij and Liens, 1993), Landscapes (Wright, 1995), and parallel coordinates visualization technique (Inselberg, 1985; Inselberg and Dimsdale, 1990). The parallel coordinate technique maps the k-dimensional space onto the two-dimensional display using parallel coordinates which are parallel to one of the display axes. Also, the axes correspond to the dimensions. Each data item is presented as a polygon line, intersecting each of the axes at that point which corresponds to the value of the considered dimension.

Another class of visual data exploration techniques is the icon-based techniques. The idea is to map the attribute values of a multidimensional data item to the features of an icon. Classes of icon-based techniques are stick figures (Pickett and Grinstein, 1988), shape-coding (star icons, Chernoff faces) (Beddow, 1990), and color icons (Levkowitz, 1991; Keim and Kriegel, 1994). In the case of the stick figure technique, for example, two dimensions are mapped to the display dimensions and the remaining dimensions are mapped to the angles and/or limb length of the stick figure. The data items are relatively dense with respect to the two-display dimensions, the resulting visualization presents texture patterns that vary according to the characteristics of the data.

The basic idea of pixel-based techniques (Keim, 1996) is to map each dimension value to a colored pixel and group the pixels belonging to each dimension into adjacent areas. Since, in general, pixel displays use one pixel per data value, the techniques allow the visualization of the largest amount of data possible (up to about 1,000,000 data values). Pixel techniques use different arrangements for different purposes. By arranging the pixels in an appropriate way, the resulting visualization provides detailed information on local correlations, dependencies, and hot spots.

Stacked display techniques are tailored to present data partitioned in a hierarchical fashion. In the case of multidimensional data, the data dimensions to be used for partitioning the data and build the hierarchy have to be selected appropriately. An example of a stacked display technique is hierarchical stacking (LeBlanc et al., 1990). The basic idea is to embed one coordinate system inside another coordinate system, i.e., two attributes form the outer coordinate system, two other attributes are embedded into the outer coordinate system, and so on. The display is generated by dividing the outermost level coordinate systems into rectangular cells and, within the cells, the next two attributes are

4. Classification of Multidimensional Visualization Techniques

Many novel visualization techniques have been developed and existing techniques have been extended to work with larger data sets and make the displays interactive. For most
V. Šaltinis, J. Anžaitė

Table 3
An attempt to compare multidimensional visualization techniques
(++) very good, (+) good, (0) neutral, (−) bad, (−−) very bad

<table>
<thead>
<tr>
<th>Geometric Techniques</th>
<th>Clustering</th>
<th>Multivariate hot spot</th>
<th>No. of attributes</th>
<th>No. of data objects</th>
<th>Categorical data</th>
<th>Visual overlap</th>
<th>Learning curve</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scatterplot Matrices</td>
<td>++</td>
<td>++</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>0</td>
<td>++</td>
</tr>
<tr>
<td>Landscapes</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>0</td>
<td>0</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Proportion Views</td>
<td>++</td>
<td>++</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>0</td>
<td>++</td>
</tr>
<tr>
<td>Hypersurface</td>
<td>+</td>
<td>++</td>
<td>-</td>
<td>0</td>
<td>0</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Parallel Coordinates</td>
<td>0</td>
<td>0</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Item-based Techniques</th>
<th>Clustering</th>
<th>Multivariate hot spot</th>
<th>No. of attributes</th>
<th>No. of data objects</th>
<th>Categorical data</th>
<th>Visual overlap</th>
<th>Learning curve</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stick Pictures</td>
<td>0</td>
<td>0</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>Shape</td>
<td>0</td>
<td>-</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td>Coding</td>
<td>0</td>
<td>-</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td>Color Code</td>
<td>0</td>
<td>-</td>
<td>-</td>
<td>0</td>
<td>0</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Proxim-oriented Techniques</th>
<th>Clustering</th>
<th>Multivariate hot spot</th>
<th>No. of attributes</th>
<th>No. of data objects</th>
<th>Categorical data</th>
<th>Visual overlap</th>
<th>Learning curve</th>
</tr>
</thead>
<tbody>
<tr>
<td>Query-Independent</td>
<td>+</td>
<td>++</td>
<td>++</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Query-Dependent</td>
<td>+</td>
<td>++</td>
<td>++</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Hierarchical Techniques</th>
<th>Clustering</th>
<th>Multivariate hot spot</th>
<th>No. of attributes</th>
<th>No. of data objects</th>
<th>Categorical data</th>
<th>Visual overlap</th>
<th>Learning curve</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dimension Stacking</td>
<td>+</td>
<td>+</td>
<td>0</td>
<td>0</td>
<td>+</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Treemap</td>
<td>+</td>
<td>0</td>
<td>0</td>
<td>+</td>
<td>+</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Cone Trees</td>
<td>+</td>
<td>-</td>
<td>0</td>
<td>0</td>
<td>+</td>
<td>0</td>
<td>+</td>
</tr>
</tbody>
</table>

used to span the second level coordinate system. This process may be repeated one more time. The usefulness of the resulting visualization largely depends on the data distribution of the outer coordinates and, therefore, the dimensions which are used for defining the outer coordinate system have to be selected carefully. A rule of thumb is to choose the most important dimensions first. Other examples of hierarchical techniques are a treemap (Shneiderman, 1992; Johnson, 1993), and cone-trees (Robertson et al., 1991).

This brief classification is aimed at providing a more structured understanding of a large number of available multidimensional visualization techniques. It can also be used as a starting point to compare the available techniques, to improve the existing ones, and to develop new techniques. Table 3 is trying to compare a number of visualization techniques. The comparison of the visualization techniques is based on their suitability for certain:

- **task characteristics**, such as clustering and multivariate hot spots,
- **data characteristics**, such as number of dimensions (attributes), number of data objects, and suitability for categorical data,
- **visualization characteristics**, such as visual overlap and learning curve.

Ripley (1996) divides statistical data-analysis methods into clustering methods, projection methods, and multidimensional scaling (MDS) methods.

5. Clustering Methods

The goal of clustering is to reduce the amount of data by categorizing or grouping similar data items together. Such grouping is pervasive in the way humans process information and one of the motivations for using clustering algorithms is to provide automated tools to help in constructing categories or taxonomies (Jardine and Sibson, 1971; Sneath and Sokal, 1973). The methods may also be used to minimize the effects of human factors in the process.

Clustering methods can be divided into two basic types: hierarchical and partitional clustering.

Hierarchical clustering proceeds successfully by either merging smaller clusters into larger ones, or by splitting larger clusters. The clustering methods differ in the rules by which it is decided which two small clusters are merged or which large cluster is split. The end result of the algorithm is a tree of clusters called a dendrogram, which shows how the clusters are related.

Partitional clustering attempts to directly decompose the data set into a set of disjoint clusters. The criterion function that the clustering algorithm tries to minimize may emphasize the local structure of the data, by assigning clusters to peaks in the probability density function, or the global structure. Typically, the global criteria involve minimizing some measure of dissimilarity in the samples within each cluster, while maximizing the dissimilarity of different clusters.

A commonly used partitional clustering method, K-means clustering (MacQueen, 1967), uses the criterion function - the average squared distance of the data items \( x_k \) from their nearest cluster centroids:

\[
E_K = \sum_{k} |x_k - m_k(x_k)|^2,  \tag{1}
\]

where \( c(x_k) \) is the index of the centroid that is closest to \( x_k \). One possible algorithm for minimizing the cost function begins by initializing a set of \( K \) cluster centroids denoted by \( m_i, i = 1, \ldots, K \). The positions of \( m_i \) are then adjusted iteratively by first assigning the data samples to the nearest clusters and then recomputing the centroids. The iteration is stopped when \( E \) does not change markedly any more. In an alternative algorithm, each randomly chosen sample is considered in succession, and the nearest centroid is updated.

Eq. 1 is also used to describe the objective of a related method, vector quantization (Gersho, 1979; Gray, 1984; Mallat et al., 1985). In the vector quantization, the goal is to minimize the average (squared) quantization error, the distance between a sample \( x \) and its representation \( m_c(x) \). The algorithm for minimizing (1) that was described above is actually a straightforward generalization of the algorithm proposed by Lloyd (1957) in a one-dimensional setting.
The interpretation of clusters may be difficult. Most clustering algorithms prefer certain cluster shapes, and the algorithms will always assign the data to clusters of such shapes even if there were no clusters in the data. Usually the goal is not just to compress the data set, but also to make inferences about its cluster structure. The results of the cluster analysis need to be validated (Jain and Dubes, 1988).

Another potential problem is that the choice of the number of clusters may be critical: quite different kinds of clusters may emerge when \( K \) is changed. Good initialization of the cluster centroids may also be crucial: some clusters may even be left empty if their centroids lie initially far from the distribution of data.

6. Projection Methods

The goal of the projection methods is to represent the input data items in a lower-dimensional space in such a way that certain properties of the structure of the data set are preserved as faithfully as possible.

6.1. Linear Projection Methods

Principal Component Analysis (PCA) is a multivariate statistical method for linearly transforming a sample of \( N \) \( p \)-variate vectors

\[
\mathbf{X} = [x_{ik}; i = 1, \ldots, N; k = 1, \ldots, p]
\]

into a new sample of \( q \)-variate vectors

\[
\mathbf{Y} = [y_{iq}; i = 1, \ldots, q],
\]

such that the columns of \( \mathbf{Y} \) are uncorrelated, and \( q \leq p \) (Flury, 1997). Each of the \( q \) derived variables is expressed as a linear combination of \( p \) correlated, measured variables (Hotelling, 1933).

The \( q \) derived variables are referred to as Principal Components (PCs), and form a system of orthogonal axes. If we consider \( \mathbf{X} \) to be a configuration of \( N \) points

\[
\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})^T
\]

in a \( p \)-dimensional Euclidean space, then the first PC defines a line through this space that minimises the sum of squared distances of the points from it, and thus maximises the variance of coordinates \( \{x_{i1} = y_{i1}\} \) of an orthogonal projection of \( \{\mathbf{x}_i\} \) on this line (Pearson, 1901). The second PC is a line that maximises the variance of the projection coordinates \( \{x_{i2} = y_{i2}\} \) of the points, subject to being perpendicular to the first PC. Taken together, the first two PCs give a plane of the closest fit to the configuration \( \mathbf{X} \), i.e., one that minimises the sum of squared distances of the points to that plane. The remaining PCs are defined recursively in this manner, and individually account for a smaller amount of the total variance. The first \( q' \leq q \) principal components give the best linear approximation.

6.2. Nonlinear Projection Methods

PCA cannot take into account nonlinear structures, consisting of arbitrarily shaped clusters or curved manifolds. Projection pursuit tries to express some nonlinearities, but if the data set is high-dimensional and highly nonlinear, it may be difficult to visualize it with linear projections onto a low-dimensional display even if the projection angle is chosen carefully.

Several approaches have been proposed for reproducing nonlinear higher-dimensional structures on a lower-dimensional display. The most common methods allocate a representation for each data point in the lower-dimensional space and try to optimize these representations so that the distances between them would be as similar to the original distances as possible. The methods differ in how the different distances are weighed and how the representations are optimized.

6.2.1. Multidimensional Scaling

MDS refers to a group of methods that is widely used especially in behavioral, econometric, and social sciences to analyze subjective evaluations of paired similarities of entities. The starting point of MDS is a matrix consisting of dissimilarities of the entities.

The goal of MDS methods is not merely to create a space that would represent the relations of the data faithfully, but also to reduce the dimensionality of the data set to a sufficiently small value to allow visual inspection of the set. There exists a multitude of variants of MDS with slightly different cost functions and optimization algorithms.

The algorithms of MDS can be roughly divided into two basic types: metric and nonmetric MDS.

In the original metric MDS (Torgerson, 1952; Young and Householder, 1938), the distances between the data items have been given, and the configuration of points that would give rise to the distances sought are obtained. Often a linear projection onto a subspace obtained by PCA is used. The key idea of the method is to approximate the original set of distances corresponding to the configuration of points in a Euclidean space. One way to construct a nonlinear projection method is to use a low-dimensional, say, two-dimensional data vector \( \mathbf{x}' \), then the goal of projection is to optimize the representations so that the distances between the items in the two-dimensional space would be as close to the original distances as possible. If the distance between \( x_{ik} \) and \( x_{il} \) is denoted by \( d(k, l) \), the metric MDS tries to approximate \( d(k, l) \) by \( d'(k, l) \). If a square-error cost is used, the objective function, called a stress function, to be minimized can be written as

\[
E_M = \sum_{k \neq l} \left[ d(k, l) - d'(k, l) \right]^2.
\]

Function (3), called a stress function, is more suitable for optimization:

\[
E_M = \sum_{k \neq l} \left[ \left( d(k, l) \right)^2 - \left( d'(k, l) \right)^2 \right].
\]
Many authors minimize different versions of the stress function by means of either local techniques or their modifications. Some of these modifications are derived to achieve non-local behavior of the method. The most widely known method of this kind is SMACOF, based on the majorization of an objective function (DeLeeuw, 1977; DeLeeuw, 1988). The tunneling method is known as one of global optimization methods (Torn and Zilinskas, 1989). It was adapted to the problems of multidimensional scaling and investigated by P. Groenen (1993).

A perfect reproduction of the Euclidean distances may not always be the best possible goal, especially if the components of the data vectors are expressed on an ordinal scale. The rank order of the distances between the vectors is meaningful only, not the exact values. The projection should try to match the rank order of the distances in the two-dimensional output space to the rank order in the original space. The best possible rank ordering for a given configuration of points can be guaranteed by introducing a monotonically increasing function \( f \) that acts on the original distances. Nonmetric MDS (Kruskal, 1964; Shepard, 1962) uses such a function (Kruskal and Wish, 1978), whereby the normalized cost function becomes

\[
E_N = \sum_{k \neq l} \frac{1}{d^2(k, l)} \sum_{k \neq l} \left[ f(d(k, l)) - d(k, l) \right]^2.
\]

(4)

For any given configuration of the projected points \( x'_k \), \( f \) is always chosen to minimize (4).

Another nonlinear projection method, Sammon's mapping (Sammon, 1969), is closely related to the metric MDS version described above. It, also, tries to optimize the cost function:

\[
E_S = \sum_{k \neq l} \frac{[d(k, l) - d'(k, l)]^2}{d(k, l)}.
\]

(5)

The only difference between Sammon's mapping and the nonlinear metric MDS (2) is that the errors in distance preservation are normalized. Because of the normalization, the preservation of small distances will be emphasized.

The principal curves (Hastie and Stuetzle, 1989) are smooth curves that are defined by the property that each point of the curve is the average of all data points that project to it, i.e., for which that point is the closest point on the curve. Otherwise, the curves pass through the center of the data set. Principal curves are generalizations of principal components, extracted using PCA, in the sense that a linear principal curve is a principal component.

7. The Self-Organizing Map Algorithm

Competeive learning is an adaptiv process in which the neurons in a neural network gradually become sensitive to different input categories, sets of samples in a specific domain of the input space (Kohonen, 1982; Kohonen, 1984; Nass and Cooper, 1975). A kind of division of labor emerges in the network when different neurons specialize to represent different types of inputs.

The specialization is enforced by a competition among the neurons: when an input \( z \) arrives, the neuron that is best able to represent it wins the competition and is allowed to learn it even better.

If there exists an ordering among the neurons, i.e., the neurons are located on a discrete lattice, a self-organizing map, a competitive learning algorithm can be generalized: if not only the winning neuron, but also its neighbors are allowed to learn, neighboring neurons will gradually specialize to represent similar inputs, and the representations will become ordered on the map lattice.

The neurons represent the inputs with reference vectors \( m_i \), the components of which correspond to synaptic weights. One reference vector is associated with each neuron called a unit in a more abstract setting. The unit, indexed by \( c \), whose reference vector is nearest to the input \( x \), is the winner of the competition:

\[
c = c(x) = \arg \min \{ ||x - m_i||^2 \}.
\]

(6)

The winning unit and its neighbors adapt to represent the input even better by modifying their reference vectors towards the current input. If the locations of units \( i \) and \( j \) on the map grid are denoted by the two-dimensional vectors \( r_i \) and \( r_j \), respectively, then

\[
h_{ij}(t) = h_{ij}(||r_i - r_j||; t),
\]

where \( t \) denotes time.

During the learning process at time \( t \) the reference vectors are changed iteratively according to the following adaptation rule:

\[
m_i(t + 1) = m_i(t) + \alpha_i(t) \{ x(t) - m_i(t) \},
\]

(7)

where \( x(t) \) is the input at time \( t \) and \( c = c(x(t)) \) is the index of the winning unit.

8. A Sample of Using Multidimensional Scaling

Although improved local search procedures are used for some applications of multidimensional scaling, certain applications can be solved only with global optimization. An example of such applications is described in (Mathar, 1996).

A frequently used test problem for multidimensional scaling algorithms (Mathar and Zilinskas, 1994; Green et al., 1989) is based on the experimental results, on testing several soft drinks (Green et al., 1989). 38 students have tested ten different Cola brands. Each pair was judged on its dissimilarity on a 9-point scale (1 = very similar, 9 = completely different). The accumulated dissimilarities are the data of the problem. The scaled down accumulated dissimilarities are given in Table 4.
The goal of this multidimensional scaling problem is to find the configuration of 10 objects, representing each Cola brand in a two-dimensional space, which would help to interpret the data. It is shown in (Mathar, 1996) that there are many local minima and, interpreting the data on the basis of the achieved configuration from local minima, leads to different results. Therefore, it is necessary to find the global minimum and the corresponding configuration which explains the data best.

The number of objects for this global optimization problem is $N = 10$. The configuration of objects in $m = 2$-dimensional space should be found. The number of variables is $n = 17$. The feasible region is $\{0, 1\}^2 \times [-1, 1]^{15}$. The objective function of the problem is (3).

The representation of the Cola problem is shown in Fig. 4.

### Table 4

<table>
<thead>
<tr>
<th></th>
<th>Pepsi</th>
<th>Coke</th>
<th>Classic Coke</th>
<th>Diet Pepsi</th>
<th>Diet Coke</th>
<th>Diet Slice</th>
<th>Diet 7-Up</th>
<th>Dr. Pepper</th>
<th>Slice</th>
<th>7-Up</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pepsi</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Coke</td>
<td>1.27</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Classic Coke</td>
<td>1.69</td>
<td>1.43</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Diet Pepsi</td>
<td>3.34</td>
<td>2.32</td>
<td>1.83</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Diet Coke</td>
<td>3.09</td>
<td>2.18</td>
<td>2.26</td>
<td>2.85</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Diet Slice</td>
<td>3.00</td>
<td>2.08</td>
<td>2.27</td>
<td>2.88</td>
<td>1.55</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Diet 7-Up</td>
<td>3.20</td>
<td>2.12</td>
<td>2.22</td>
<td>2.88</td>
<td>1.55</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dr. Pepper</td>
<td>2.86</td>
<td>2.56</td>
<td>2.56</td>
<td>2.54</td>
<td>2.99</td>
<td>3.12</td>
<td>3.09</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Slice</td>
<td>3.15</td>
<td>2.18</td>
<td>2.18</td>
<td>2.12</td>
<td>1.34</td>
<td>1.64</td>
<td>3.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7-Up</td>
<td>3.21</td>
<td>2.18</td>
<td>2.18</td>
<td>2.12</td>
<td>1.70</td>
<td>2.56</td>
<td>2.95</td>
<td>1.32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tab</td>
<td>2.38</td>
<td>2.31</td>
<td>2.42</td>
<td>1.94</td>
<td>2.85</td>
<td>2.81</td>
<td>2.56</td>
<td>2.91</td>
<td>2.97</td>
<td></td>
</tr>
</tbody>
</table>

Fig. 4. The solution of the Cola problem.

9. Visualization in Education

Education can be viewed as the externally facilitated development of knowledge. This external influence can take many forms (a teacher, textbook, article, movie, TV show, computer program...). The purpose of any visualization to be used in an educational context is to facilitate the learning of some knowledge (idea, concept, fact, algorithm, relationship...). In order to accomplish this, visualization should relate knowledge the learner has and the knowledge being taught. Therefore, in order to design effective visualizations, it is necessary to know (or at least have a theory about) what the learner knows.

The purpose of graphical display is to provide the viewer with a visual means of processing the information. It is important to note, that for a visualization to be effective it must make use of the knowledge base of the viewer. If the viewer does not possess the knowledge to understand the graphical entities and relations between them, then the visualization will not achieve its goal. For the most part visualization techniques can be classified into two categories:

- Data exploration,
- Communicating information.

Data Exploration is the practice of using visualization techniques to find unforeseen relationships between data points or sets of points in large databases. Once a relationship has been found, the same visualization can be used to communicate that relation to others. Visualization techniques can also be applied to information that is already known.

In their work (1992), Hiebert and Carpenter present a framework for discussing the representation of mathematical concepts in the context of teaching for understanding. The concepts in the framework are not specific to mathematics and can be applied to the representation of knowledge in general. Their framework also provides a useful means for discussing the role that external representations play in learning. The main points of their framework are:

- relationships exist between external and internal representations;
- the form of the external representation with which the student interacts influences how knowledge is represented internally;
- the form in which a student externally represents knowledge if influenced by their internal representation;
- internal representations of knowledge are connected to form networks of knowledge;
- networks can be hierarchical – some representations subsume others;
- networks can be graph-like – the nodes represent pieces of information and the arcs represent the relationships between them;
- most likely, a combination of the two of them;
- understanding occurs when an idea is well integrated into a richly connected network.
10. Conclusions
In this paper, we have presented an overview of major visualization techniques. Each of these techniques is based on distinct concepts and produces distinct images. No optimal visualization method can be selected. The variety of visualization methods exists because different data and different practical problems require a corresponding visualization technique. The overview is, of course, incomplete and does not include, for example, incremental MDS. Gathering of visualization methods and techniques lets us to investigate a new method or technique.

References

Data Visualization: Ideas, Methods, and Problems
Duomenų vizualizacija: idėjos, metodai, problemas

Vydūnas ŠALTENIS, Jūratė AUŠRAIČĖ


Bertino teorija apie įtvirtinimą M. Greens. Jis teigia, kad ėmaitė daugiau net daugiau išvaizdų kintamųjų įrašų, aplink kurias Bertinas nežinome ir, kad trijų komponentų vizualizacijos yra nebūtini ėlektyviniai (gali būti kurių ir daugių). Duomenų ėmaitė būtinai tokiais kintamaisiais, kaip judesys (motion), nuolatinius (flecker) bei bendravimą (spons).

Kalbant apie duomenų vizualizavimą, svarbu atsižvelgti į vieną iš tris duomenų įrašų. Toks atitinka kitą vieną vizualizacijos būdą: ikemės į skirtingus ir geometrinius projekcijos, piktogramines, taikinių (pixel-based), hierarchinės (hierarchical), o taip pat bandome juos jungti.

Aprašyta keturi populiariausi duomenų vizualizavimo metodai: klasterinių (clustering), projekcijos (projection), daugiamačio skaičiaus (multidimensional scaling) bei save organizuojantį žemėlapių (self-organizing maps). Visualizavimo metodai gali būti skirtingi – tiek savo sprendimo idėjos, tiek gauti vaizdai.