Andreas Loizides

Supervisor: Prof. Mel Slater

Department of Computer Science,
UCL University of London,
Gower Street,
London WC1E 6BT, UK.

1 Methodology

Here, we will present the formal theory behind the Empathic Visualisation Algorithm (EVA). The method is based on a technique devised by Slater [?] to automatically map data to visual structures. First, we will present a statement of the research problem.

1.1 Statement of the problem

The problem we are dealing with is that of Empathetic Visualisation. This is a novel technique for the Interactive Visualisation of Complex Data in naturalistic form.

By complex data set we mean data sets that are relatively large both in terms of the amount of data there is and the number of variables (dimensionality) that the data encompasses.

The most interesting data is multi-dimensional. How can this be represented on a graph? The data is large in the sense that it cannot be viewed and understood by conventional statistical techniques. Moreover, only a relatively small class of problems are directly amenable to statistical analysis - those where the variables are understood, potential relationships are already known, and there are hypothesis to be tested. Numerous techniques have been described in the literature that attempt to visualise such data with their advantages and disadvantages also being described.

Here we are considering both cases of data that are directly amenable to statistical analysis and cases of data where even the questions to be posed on the data are not fully formulated. A classic example described in the proposal of this thesis, is that of financial, say, stock exchange data over a number of years - the movement of share prices, for example. Here the question might be *"What determines the changes in share prices?"* Such a vague question is not yet ready for statistical analysis - there are a huge number of potential variables (economic indicators, social conditions, the interrelationships of world stock exchange movement), and the amount of data itself is massive (over how many years?).

In such situations information visualisation helps in "understanding" a set of data (which may be dynamically unfolding in time) by allowing users to visualise representations of the data, thus using vision to build "understanding", and allowing the formation of hypothesis for later statistical analysis.

There are two main problems involved in this:

- The choice of a suitable mapping from the data to the chosen visual representation, structure. The difference between arbitrary and automatic mapping. Investigating the possibilities of taking into account the emotions of the user of the visualisation in an automatic mapping, is at the heart of the problem.

- The choice of a suitable paradigm for the representation of the data (the classic stock exchange portray shares as say wheat, the height of the wheat as the price, the wind as the "economic climate", and so on). The choice of a suitable paradigm, whether abstract, or more realistic, and the flexibility to dynamically change the representations is at the heart of the problem - different representations may lead to quite different understandings of the data. The possibilities of naturalistic representations is also at the heart of the problem.

For this PhD we have in mind the investigation of an automatic mapping paradigm on a naturalistic visual representation. For reasons that should become apparent below we call this *"Empathetic Visualisation"*. Figure 1 shows the placement of EVA on the classification proposed earlier.

The specific goal of the project will be to construct a system for such a representation, and then to test this in an experimental setting. The system

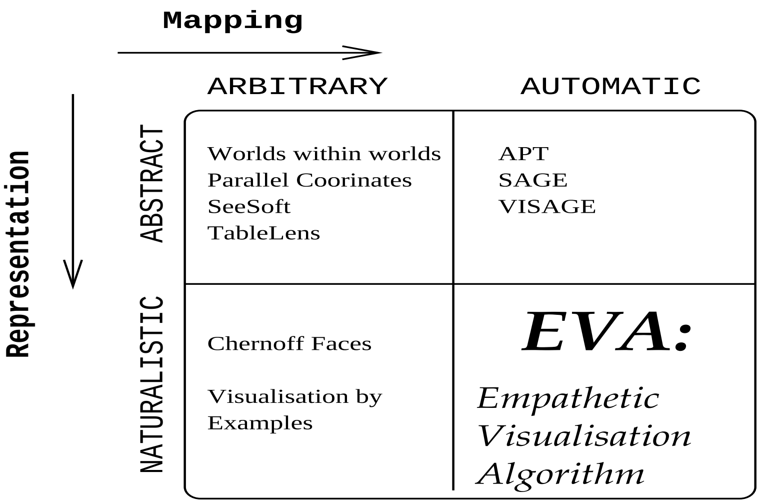

Figure 1: Classification of EVA,

should be such that it can be used with as different data sets as possible- i.e. that is a generic rather than tied to a particular form of data.

1.2 Fundamentals

Throughout this thesis we are considering the representation of multivariate data in an $n \times k$ data matrix X, consisting of n cases on k "quantifiable" variables $x_1, x_2, ..., x_k$. Each row in the data matrix typically represents an individual and there are k observations made for each individual. The objective is to construct a visualisation of the data matrix such as the salient features of the data can intuitively by recognised by an observer and representation gives an overall view of the data set. Within this overall objective there are two further objectives that were mentioned earlier and are fundamental:

1. Naturalistic visual representation. It should be something encountered in everyday life, something that does not require special knowledge for interpretation by a normal human observer.

2. Automatic mapping. The mapping should be that semantically "important" features in the data are mapped to "important" features of the visual structure that are significant to human perception or emotion.

Examples of (1) include faces, buildings, towns and other. No human needs to be an expert to recognise the emotional content of another human face - it is recognisably "happy", "sad", "angry", "relax", "fear", "neutral" and various combinations of these basic emotions. We frequently use, in this project, an example of a "face" because it is the epitome of a naturalistic visual structure in the sense we mean here.

Given a set of data, and given a visual structure, it is trivially easy to construct a mapping from the data to a visual structure - for example map variable x_i to the ith facial feature. However, such a mapping is arbitrary - it does not take into account the impact of the face on the emotions of the observer. Now the data is of interest to the observer for some reason: associating with the data there will be some "value system" reflecting the interest or importance or consequences of aspects of the data for the situation of the observer. A fundamental goal, reflected in (b) is that the perceptually or emotionally significant features of the visual structure directly reflect the value system over the data. We call this *visual homomorphism.* Hence the mapping from data to visual structure cannot be arbitrary, but must be constructed in such a way that this visual homomorphism is realised.

What follows is an explanation of how such a mapping can be constructed. We consider two different types of visualisation problem: the first is the representation of the data matrix as a whole - one visual structure represents the entire data matrix i.e. a face represents the entire data set. The second, is where each row (i.e. individual) in the data is to be represented by a different instance of the same type of visual structure - so using faces again, each row is mapped to a different face. In the first case, the problem is to capture overall features of the same data set. In the second case the problem is to examine the differences in the individuals, or to examine one individual changing through time. In fact the method is very similar in both cases - the difference is on the focus and interpretation. We consider the overall representation first.

1.3 Assumptions and Notation

Let $\nu_s(X)$, $s = 1, 2, ..., p$ be p functions over the data representing "values" over the data matrix.

Consider the example where the data matrix represents a set of customers of a telephone company, and the variables are quantities such as age, gender, marital status, income, number of years with the company, number of telephone calls made per week, monthly phone bill, and so on. One value might be a function of the overall age distribution of the population - such as the mean age, or the percentage over 65, or the percentage under 20. Another value might be the "flatness"of the data - for example the ratio of the variance of the first principle component to the total variation in the data. Another value might be the correlation coefficient between age and monthly bill, and so on, for many other quantities that characterise the interests or "value system" of the observer.

In addition, consider a visual structure (Ω). Similarly there are p importance aspects, characteristics, of the Ω that are measurable and significant to human perception or emotions, $e_s(\Omega)$, $s = 1, 2, ..., p$. In the example of the face these might be the degree of anger, happiness, boredness, - or characteristics such as age, beauty, gender and so on.

The fundamental goal, in terms of any X and any Ω, is to produce a mapping $\mu(X) \longrightarrow \Omega$ such that 'values' over the data matrix are reflected in characteristics of the visual structure. In particular that $e_s(\Omega)$ is a monotonically increasing function of $\nu_s(X)$.

A *characteristic* is a measurement of some aspect of Ω as a whole (such as the emotions on a face) rather than some individual *feature* (such as the shape of the mouth). It is some measure representing the totality of the face i.e. the degree of "happiness". The appearance of "happiness" depends on many different individual features of the face - specific configurations of muscle tensions. Similarly, the appearance of the beauty or age or gender is derived from many different features - such as size of the eyes, inter-ocular distance, shape of the mouth, symetry, and so on. In other words, features are the individual components that make up a face - such as the specific configuration of muscle tensions for a specific face, or the geometric and material properties of the actual features (eyes, colour of the eye, mouth, nose, lips) that make up a face. Therefore, we can *render* a face knowing its features. Once rendered the face will have a set of measurable characteristics.

Suppose that there are r features of the visual structure: $\phi_t(\Omega)$, $t = 1, 2, ..., r$. Knowing these features of Ω results in being able to render it. Once rendered, we can measure it to determine its characteristics $e_s(\Omega)$, $s = 1, 2, ..., p$.

Finally, we introduce *feature functions* over the data matrix: $f_t(X)$, $t = 1, 2, ..., r$. These functions completely determine the features of the visual structure, in fact

$$f_t(X) = \phi_t(\Omega),\ t = 1, 2, ..., r \tag{1}$$

The values of these functions are *interpreted* as the values of the features of the visual structure. Our goal is to choose these functions f in order to attain the required correspondence between the value system over the data and the characteristics of the visual structure and therefore, to attain the visual homomorphism.

Let $\nu = (\nu_1, \nu_2, ..., \nu_p)$ and $e = (e_1, e_2, ..., e_p)$. Suppose $\| \nu - e\|$ is a measure of the 'distance' between these two vectors. Then the specific goal is to choose $f_t(X)$, $t = 1, 2, ..., r$ such that $\| \nu - e\|$ is minimised.

1.3.1 Using Genetic Progamming (GP)

The minimisation problem introduced above can be tackled with a GP. Let $F_i = (f_{1i}, f_{2i}, ..., f_{ri})$ be a specific set of feature functions which when applied to X produce the visual structure features. Choose at random a large collection of such sets of functions F_i, $i = 1, 2..., N$. This defines a *population* of sets of feature functions. The ith member of the population produces a specific visual structure Ω_i. This visual structure has characteristics $e_s(\Omega_i)$, $s = 1, 2, ..., p$. These characteristics can be used to produce the distance measurement $\| \nu - e\|$. The distance measurement can be used to compute a "fitness" for the ith member of the population. Hence, each member of the population has an associated fitness, which can be expressed as a probability. These probabilities determine survival into the next generation and selection for mating - thus producing a second generation. The process continues until (possible) convergence. The most fit member of all populations is chosen for the required mapping.

What would be expected is that at each successive generation, the average fitness would increase, until a generation is reached such that subsequent iterations produce only negligible increments in fitness. The most fit member of that population is chosen for the required mapping.

The basic theory of GPs will be described in the next chapter, to give a better understanding of how this optimisation function is realised in this method.

Visualising Individuals in a Data Matrix The method above produces a visualisation for an entire data matrix. Instead the observer may be interested in visualising each individual(row) of the data matrix - in order to look for "special" individuals (select companies to add to his investment portfolio), or where the rows represent the changing of one individual through time, representing an evolving situation (showing the changes in company shares during time).

The method is fundamentally unchanged, and only involves a reconsideration of the domain of the feature functions f. Before, the domain of these functions was the whole of the data matrix X. Instead now let the domain be over the variables represented by the columns of the data matrix. So each individual row of the data produces a set of feature function values, which therefore determines a visual structure for each row.

Similarly, restrict the domain of the value functions ν also to the variables represented by the columns. Hence each row now produces a distance measure - that between the characteristics of the visual structure for that row, and the values over the data for that row. The overall distance for the ith member of the population of sets of feature functions can therefore be taken as a combination (e.g., a sum) of the distances over all rows of the data. Then method then proceeds as before.

Visualising a Single Number It is interesting to note that the method works if the entire data matrix consists of one single number! Suppose this number is x. Now over x we define the value functions as usual. For example, suppose x is a parameter representing the functioning of some system (e.g., temperature). The exponential of x might have a particular and important meaning, the reciprocal of x might have another meaning, the sin of x might have another meaning, and so on. Now a first generation of random feature functions can be defined as usual, and the corresponding visual structure determined, the characteristics measured and compared to the value system. The GP would be just as in the general case. Of course, visualising a single number is not that interesting. However, a very important note that can be derived here, is that complexity of the method and the visual structure does not increase as the dimensionality of the variables increases or the actual amount of data. Throughout the literature only parallel coordinates claim to tackle this problem effectively. In their method complexity increases in a linear way compare to the variables of the data set.

1.4 Overview

A summary of the steps required in order to implement the method is described below, using the notation above. Figure 2 gives a graphical visualisation of the overview of the method.

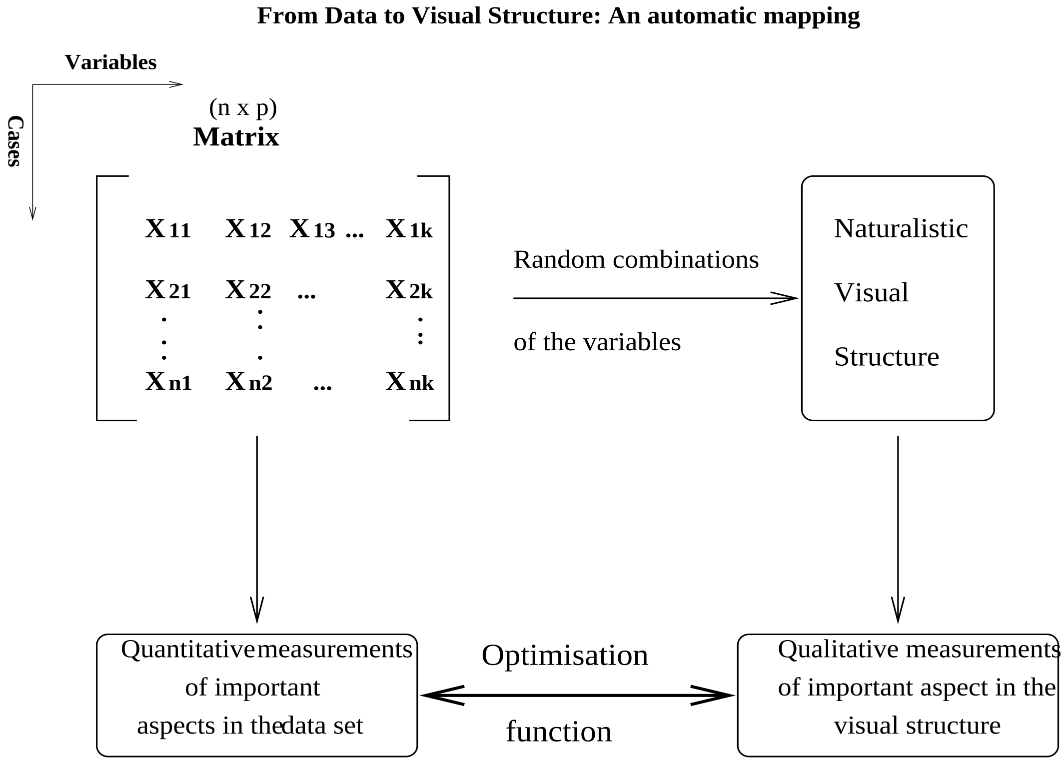

Figure 2: Overview of the method.

1. Decide on Ω, the visual structure. Determine the number of features Ω has. Assuming there are r you need $f_t(X)$, $t = 1, 2..., r$ feature functions in order to render an individual.

2. The observer identifies the p important values of the data set $\nu_s(X)$, $s = 1, 2, ..., p$.

3. Identify $e_s(\Omega)$, $s = 1, 2, ..., p$ the p characteristics of the visual structure that measure its totality and are significant to human emotions and perception.

4. Identify the fitness function i.e. the function we are trying to minimise, which is, for example:
$$\sum_{d=1}^{n} \sum_{s=1}^{p} (\nu_{ds} - e_{ds})^2 \tag{2}$$

5. Identify the GP parameters and run the GP.

The steps shown above achieve the visual homomorphism described earlier. In other words, it can be defined as the 'extraction and visualisation of qualitative data from quantitative one'.

1.5 Discussion

This section has presented a method for automatic determination of a mapping from data to visual structure. It requires a user (someone interested in the data) to construct a set of value functions of interest over the data. The designer of the visualisation must decide on a type of visual structure, and a set of perceptually or emotionally significant characteristics of this visual structure that matches the number of value functions inserted by the user. The visual structure must be determined by a set of quantifiable features. A GP is then used to construct the mapping that minimises some given measure of distance between what the user inserted and what the designer provided. The minimisation criterion is the only thing that specifies the exact nature of the mapping.

There are two assumptions that we are taking into account for the method to work:

1. GP will converge.

2. Observer will pick up out important features of the data from such mapping.

In order for the method to be of any use it will have to be tested. Before showing early testing of the method we will give a review of the field of GP.