Defining Like-minded Agents with the Aid of Visualization

Penny Noy and Michael Schroeder

City University, London EC1V 0HB, UK
{p.a.noy, msch}@soi.city.ac.uk

Abstract. Profile carrying agents offer the opportunity of meeting like minds and increasing efficiency in many information search applications. Profiles can also increase the sophistication of relationships and interactions in multi-agent systems in general. Such profiles (feature lists) may be of the agent owner (interests) or of the information sought (specifications). At the same time there is a continuing increase of profiling data of increasing complexity becoming available.

The paper first describes various possibilities for defining profiles and selecting similarity metrics in the agent interaction context. General and domain-specific data sets are specially constructed and used to provide a concrete view of the behaviour of the metrics with visualization.

Visualization usually results in xy (or xyz) coordinates for each agent, placing them in a profile space. The use of these coordinates as a means of carrying and comparing one's profile without revealing it to other agents is proposed.

Thus this paper extends our previous work on visualizing multivariate data and proximity data in the agent scenario in two ways where visualization is not the end product: using visualization tools in the phase of similarity metric choice; proposing the use of coordinates as a profile. The paper seeks to illustrate, with these applications, an objective of visual data mining, namely the increased integration of visualization and analytic techniques.

1 Introduction

Every second that passes witnesses millions of people searching for information via a myriad of means. Increasingly such means and media employed are electronic. Many people are seeking similar information and work on similar problems. Software agents carrying our profile can meet with the representatives of others and exchange important information or provide introductions. Consider the following messages from some of my (hypothetical) personal agents:

myNetworkAgent: Good morning. Professor Blake in Helsinki has just started work on one of your main research areas.

myShoppingAssistant: Here are the fourteen houses closest to your ideal specification.

myInformationSearchAgent: Here are the four most relevant documents in your specified area 'agent matchmaking' from the survey point of view.

These agents may be providing real-time search responses or background information monitoring. The concepts are embodied in many search and classification applications. These may be ordinary searches, not explicitly involving other agents (or agents at all), but agents may assist in improving precision and speed.

Of course the above scenario is far fetched in several senses: the degree to which information is searchable (i.e. specified according to agreed ontologies); the accuracy of the metrics in matching seeker with sought; the willingness of humans to allow an agent to carry their profile (and the security issues involved); privacy of information.

The focus of this paper is the task of 'matching seeker with sought'. This may involve a user profile or a profile of a task or desired piece of information. It is in this general sense that the word profile is used here. To compare profiles a metric is needed. There are many examples of this in agent systems. For instance: Faratin et al use the maximum value distance in making negotiation trade-offs[10]; The Yenga system uses correlation to determine user interests and then a direct comparison to find a common joint interest [4]; Somlo and Howe use incremental clustering for profile maintenance in information gathering web agents based on term vectors of documents the user has shown interest in [14]; GRAPPA (Generic Request Architecture for Passive Provider Agents) is a configurable matchmaking framework which includes demand and supply profiles and has been applied to matching multidimensional profiles of job applicants to vacancies [18,17]; Ogston and Vassiliadis use minimal agents working without a facilitator to match services and providers [9]. The many possible application areas divide broadly into two areas - matchmaking (e.g. matching services to clients, people connector) and search. The general topic lies within organizational concepts in multi-agent systems and improving learning with communication [21]. The *finding* and *remembering* of appropriate, like-minded agents can be centralized, left as a diffusion process or engineered in a computational ecology sense [5,4,9].

In the agent domain, as in others, visualization is often seen as a separate application that one adds on to an application for a variety of purposes. It may be of value from this point of view, but it can do more. Visual datamining seeks, amongst other things, to give the human visual system a more central role in the knowledge discovery process, to increase the integration of visualization and datamining techniques. This can be approached in a variety of ways, taking advantage of the human visual system's pattern recognition abilities, for instance, or presenting overviews of large amounts of data in novel ways. Related to this, but with a different focus, is another way, which the work presented here seeks to illustrate: the merging of the visualization and analytic processes. Here the question of whether visualization is the end result is not so important. Two examples of this are presented here:

- Overview understanding and similarity metric choice merge: Metrics are used for layouts involving dimension reduction, but different metrics produce different layouts. A metric (or other transformation process) may be needed for layout, especially in creating an overview of large complex datsets, but how is the user made aware of the different possibilities and their implications? At the same time many applications use a similarity measure - how can designers choose appropriately?
- Viewer and computational object (an agent) merge: Our layout creates a topic space. If such topic spaces are valid (they are usually approximations), can we

use this notion of spaces (or surfaces) for software agents to use when meeting or seeking other agents? In this case the visualization concept is being used to assist in agent-orientated computation. Can this help in the pursuit of the use of visualization techniques for reasoning (diagrammatic reasoning [6])?

Our earlier work looked at proximity data and multivariate data in the agent domain and indicated possible metric choices for visualization[11,13]. From the agent point of view, the question is how can we apply this and how can we assist in the problem of metric choice for profiling and classification in the agent domain. How can we meaningfully identify like-minded agents and then put this to use?

The agent paradigm is considered by some to be a valuable way of looking at problems and therefore of general application. Our work in the agent and visualization fields seeks to use visualization to serve the agent community, but, from the agent-orientated computation point of view, suggests other uses of agent ideas within visualization.

The paper first briefly surveys visualization possibilities for different types of data matrices and presents definitions of profiles, considering also the desirability of comparing profiles without revealing them. It then looks at metric choice and suggests strategies to improve choice using visualization techniques and the designing of a classification system for evaluating the metrics. The idea of using the visualization position coordinates as a profile is introduced and an example given. The nature of the examples in this paper is illustrative and the intention is to show the merging process, where visualization is not necessarily the end product, as well as to present the two specific applications.

2 Defining Profiles

In general an agent's profile is considered to be a vector of interests and behaviours (a feature list) or a similarity measure or sets and/or combinations of these [11,13]. The purpose of our work in visualization was to find layouts (in 2D or 3D) which would satisfy (usually approximately) these data either by using mathematical transformations (effective reductions via e.g. Principal Component Analysis (PCA) or distance metrics followed by Principal Coordinates Analysis (PCoA), spring embedding [3] or Self-organizing Map (SOM)[16]) or novel representations (e.g. colour maps, hierarchical axes, 'Daisy', parallel coordinates[1,2,15]).

PCA, SOM and PCoA are described briefly here as they are used in the discussion that follows:

- Principal Components Analysis: PCA is a means by which a multivariate data table is transformed into a table of factor values, the factors being ordered by importance. The two or three most important factors, the principal components, can then be displayed in 2D or 3D space.
- Self Organizing Map: The SOM algorithm [16] is an unsupervised neural net that can be used as a method of dimension reduction for visualization. It automatically organizes entities onto a two-dimensional grid so that related entities appear close to each other.

– Principal Coordinates Analysis: PCoA is used for proximity data, finding first a multivariate matrix which satisfies the distances, then transforming this into its principal components so that the two or three most important factors can be displayed in a similar fashion to PCA.

These visual representations may provide meaningful clusters or reveal patterns from which knowledge can be gained. A key problem in this area is that different methods produce different clusters (and cluster shapes). The determination of an appropriate metric [1] is a difficult problem for which general solutions are not evident. We propose the use of constructed data in a process called *signature exploration* [8] to assist in this area. This process uses specially constructed data sets to increase the user's understanding of the behaviour of visualization algorithms applied to high dimensional data.

Two developments suggest themselves from aspects of our previous visualization work: a tool for metric choice; the use of layout coordinates as a profile.

– Tool for metric choice: Agents need to compare profiles, i.e. when they meet they need to be able to compare themselves (or their tasks) and get a measure of similarity which they can interpret. Assuming (for the moment) that they are carrying their profile with them, they will need to apply an algorithm to calculate a similarity measure by both submitting their profiles to the algorithm, either both independently of the other, or via an intermediary. In designing a specific application a decision (by the designer) needs to made about what similarity measure is appropriate. The tool for metric choice developed in application of the principle of signature exploration provides an interactive interface which can help the designer to choose the metric.
– Use the layout coordinates as a profile: For layout on the screen, the data transformation or set of similarity measures results in x/y (or x/y/z) coordinates for each entity. For complex data this usually involves a significant error (i.e. it is normally not possible to find a layout which will satisfy the similarity measurements - on the one hand - and matrix transformations and truncations to 2 or 3 attributes rely on a sharp fall off of the relevant eigenvalues, which is unusual for complex data sets - on the other hand). Nevertheless such algorithms are commonly used and thus the approximations involved are often adequate. The relevant point here is that the end result is that there is an x/y (or x/y/z) coordinate pair associated with each entity and *within the current space of possibilities* this locates their *interest position*. This suggests the possibility of them carrying a much more lightweight *position profile* with them, that also means they can compare positions without revealing profiles.
  The use xy or xyz coordinates as the profile avoids revealing the profile, but the implication is that either there must be a central entity which will do the calculation (and thus that one needs to reveal one's profile to) and then give the agent its coordinates and the bounds of the space (so that it can judge relative similarity). Also this does not deal easily with dynamic situations (i.e. reflecting changing profiles), as it would require a periodic return to base to profile updating. A possible alternative is to calculate one's own coordinates with respect to a number of reference

[1] metric is here used in a general sense to mean a means of measurement which may not result in a numerical measure directly, i.e. possibly indirectly by means of layout position derived directly from SOM or PCA

points, i.e. calculate one's proximity to the reference points and then find a position in space to satisfy this reduced set of distances.
For instance for a feature list of length 5, consisting of a set of five possible agent interest areas and interest values in the range 0 to 1 (say), the following is an indication of the bounds of the space.

	A	B	C	D	E
agent1	1	0	0	0	0
agent2	0	1	0	0	0
agent3	0	0	1	0	0
agent4	0	0	0	1	0
agent5	0	0	0	0	1

It may be unwise to base the position on a computation that satisfies the similarity measures to all of these vectors (since this increases the inaccuracy of the layout), but the agent could carry the set of coordinates for certain bounds (or other reference vectors) and profile position, having the calculations made back at base. These ideas are illustrated below.

3 Choosing a Metric - Possibilities

For specific applications different metrics are used, this means that often an applications area uses one metric only. Measures may be chosen because of time complexity issues, rather than that they provide the most accurate or appropriate measure. There is also a link between the creation of the feature list and the metric choice (i.e. the formulation of the feature list affects which metric provides the most appropriate clustering) which is a further complication. In general terms the choice of metric and creation of the feature list should correspond to the required classification, but in many situations the starting point is an unknown set of data and clustering indications are sought. There is no training set and no classification. It is likely that different classifications exist. In fact there are hidden classifications, that is to say, the user has a set of things they are interested in and they would like to have the entities (other users, documents..) classified according to these groupings. One of the purposes of the signature exploration process that is being developed is to explore the mapping to clusters (via various metrics) of features of interest to the user. Originating as part of work to increase comprehension and choice of algorithms for visualization of complex data, it does not focus on feature list construction but on metric choice for a given feature set. In the process of constructing data sets for evaluation of the different options the user creates an ad hoc classification system for assessment purposes (demonstrated below).

3.1 How to choose - metrics, feature selection and weighting

The first issue is to specify the variables to be used in describing the profile and the ways in which pairwise similarities can be derived from the matrix formed by the set of profiles.

Many different measures of pairwise similarity have been proposed [7,19]. Some are closely related to one another. Measures are usually presented that are particularly relevant for comparing objects that are described by a single type of variable. This discussion restricts itself to quantitative data type for brevity.

Quantitative variable Let x_{ik} denote the value that the kth quantitative variable takes for the ith object ($i = 1,\ldots,n;\ \ k = 1,\ldots,p$). The Minkowski metric defines a family of dissimilarity measures, indexed by the parameter λ.
Minkowski metric

$$d_{ij} = (\sum_{k=1}^{p} w_k^{\lambda} |x_{ik} - x_{jk}|^{\lambda})^{1/\lambda} \ \ (\lambda \geq 1) \tag{1}$$

where $w_i (k = 1,\ldots,p)$ are non-negative weights associated with the variables, allowing standardization and weighting of the original variables. Values of λ of 1 and 2 give the two commonly used metrics of this family.
City block

$$d_{ij} = \sum_{k=1}^{p} w_k |x_{ik} - x_{jk}| \tag{2}$$

Euclidean distance

$$d_{ij} = (\sum_{k=1}^{p} w_k^2 (x_{ik} - x_{jk})^2)^{1/2} \tag{3}$$

These measures can be standardized, for instance so that d_{ij} is bounded by 1. If $w_k = (p\mathcal{R}_k)^{-1}$, where $\mathcal{R}_k$ denotes the range of values taken by the kth variable. One could also consider $w_k = p(\max_{k=1}^{p}(\mathcal{R}_k))$, which preserves the quantitative comparison between objects. Also consider not the range, but (assuming it is relevant to consider the possible minimum value then take $w_k = p(\max_{i=1}^{n}(x_{ik}))$ or $w_k = p(\max_{k=1}^{p}(\max_{i=1}^{n}(x_{ik}))$. For example:-

	Sport	Art	Music
agent1	5	1	3
agent2	4	1	5

Without weighting this gives 1.29, with the range 0.816, with max value 0.086.

Sometimes it is the relative magnitudes of the different variables that is of interest - the behaviour across the variables rather than the absolute values. Put another way, the variables describing the object define a vector with p components and interest is in the comparison of the directions of the vectors. In the following metric the cosine of the angle between the vectors is used. Since values are between -1 and 1, the measure can be transformed to take values between 0 and 1 by defining $s'_{ij} = (1 + s_{ij})/2$.
Angular separation

$$s_{ij} = \frac{\sum_{k=1}^{p} x_{ik} x_{jk}}{(\sum_{k=1}^{p} x_{ik}^2 \sum_{l=1}^{p} x_{jl}^2)^{1/2}} \tag{4}$$

For the previous example this metric gives a value for s' of 0.0465.

Mixed variables For profiling of objects the variables will sometimes be of different types: for example a person can be described in terms of their gender (binary variable), their age (quantitative variable, their amount of interest in a subject (ordinal variable if sectioned quantitative variable is used) and their personality classification (nominal variable). A general measure is:-
General similarity coefficient

$$s_{ij} = \sum_{k=1}^{p} s_{ijk}; \;\; d_{ij} = \sum_{k=1}^{p} d_{ijk} \tag{5}$$

where s_{ijk} (and correspondingly d_{ijk}) denotes the contribution to the measure of similarity provided by the kth variable. The values of s_{ijk} and d_{ijk} can take definitions as appropriate to the variable type.

Selection (feature extraction) and standardization (normalization) Sometimes it is clear what variables should be used to describe objects. In our case, with profiling queries, documents, specifications and personal profiles, it is likely that variables have to be selected from many possibilities. Thus the process is not straightforward. The pattern recognition literature describes the appropriate specification of variables as feature extraction. It is tempting to include a large number of variables to avoid excluding anything relevant, but the addition of irrelevant variables can mask underlying structure. Whilst the choice of relevant variables is important, there is also the possibility (particularly here - multidimensional nature of profiles themselves) that there is more than one relevant classification based on different, but possible overlapping, sets of variables.

Having determined appropriate variables, there is then the question of standardizing and/or differentially weighting them, followed by the construction of measures of similarity.

One aspect to the standardization is that two variables can have very different variability across the dataset. It may or may not be desirable to retain this variability. Standardization may also be with respect to the data set under consideration or with respect to a population from which the samples are drawn. In the case of quantitative variables, standardization can be made by dividing by their standard deviation or by the range of values they take in the data set. The idea of standardization lies within the larger problem of the differential weighting of variables.

4 Choosing a Metric - Visual Exploration

To assist the process of metric choice the use of specially constructed data sets in an exploration of the algorithm behaviours is proposed in signature exploration. Thus, by examining known data we gain a concrete idea of the behaviour of the various possible metrics. We have suggested a number of possible constructed data types[8]: generic(provided by the application to illustrate the behaviour of the particular algorithm); constructed(determined by the user to illustrate the behaviours in the data that are of interest to them, for evaluation purposes this represents an ad hoc classification); query (by visualization or sql-type, based on an unknown dataset, to examine clustering

of metric in practice); landmark (to provide marker entities in the visualization); feedback (the means to enable the user to enter their assessed similarities and find or modify the appropriate metric). This paper limits itself to the first two, generic and constructed, to illustrate.

4.1 Using generic data sets

Generic data sets are those considered to illustrate the behaviour of the visualization algorithms. Simple data sets do not always give an intuitive placement after such transformations. In this examination a small matrix of 7 agents were given a randomly assigned level of interest (of 1 to 10) in 7 topics.

Agent1	9	3	4	6	5	5	5
Agent2	1	10	10	1	7	2	0
Agent3	4	1	6	8	0	5	7
Agent4	2	7	8	4	0	2	0
Agent5	3	6	4	7	1	10	6
Agent6	1	7	6	5	0	2	0
Agent7	8	1	7	1	2	5	9

Subsequently three other agents were added to illustrate (a) interests identical to agent1 but scaled, (b) agent1 with the same level of interest in each topic and three other agents as in (a), (c) two of the agents showing reverse behaviours of another two. These data sets were visualized with various distance measures (using the tool SpaceExplorer [11,13,12])and comments noted. The results illustrate the similarity in behaviour of the metrics, whilst indicating the differences obtained with the two basic types - Euclidean and Angular Separation. The measures used were Minkowski ($\lambda = 3$), City, Euclidean and Angular Separation (equations 1-4). These were followed by Principle Coordinates Analysis to find points in 2D-space that satisfied the distances. Note that the accuracy of such layouts for visualization is an issue, since it is often very low. In the case of PCoA, the eigenvalues can be examined - the sum of the values of the first three (for 3D layout) being above 70% of the sum of all the eigenvalues accounts for 70% of the variance in the data and is thus an encouraging indicator.

As an illustration of this process, figure 1 shows the three shots of City, Euclidean and Angular separation with agents a,b and c having scaled interest distribution of agent1.

4.2 User-constructed data sets

Here the user constructs data sets specific to their application, explicitly or implicitly creating a classification system with which to measure the performance of the metrics in clustering their interest feature(s). This may provide a distance matrix for comparison, or such a matrix may be obtained by an informal assessment. This could be followed by feedback analysis to obtain weightings of the feature list, but here the focus is on metric choice rather than modification.

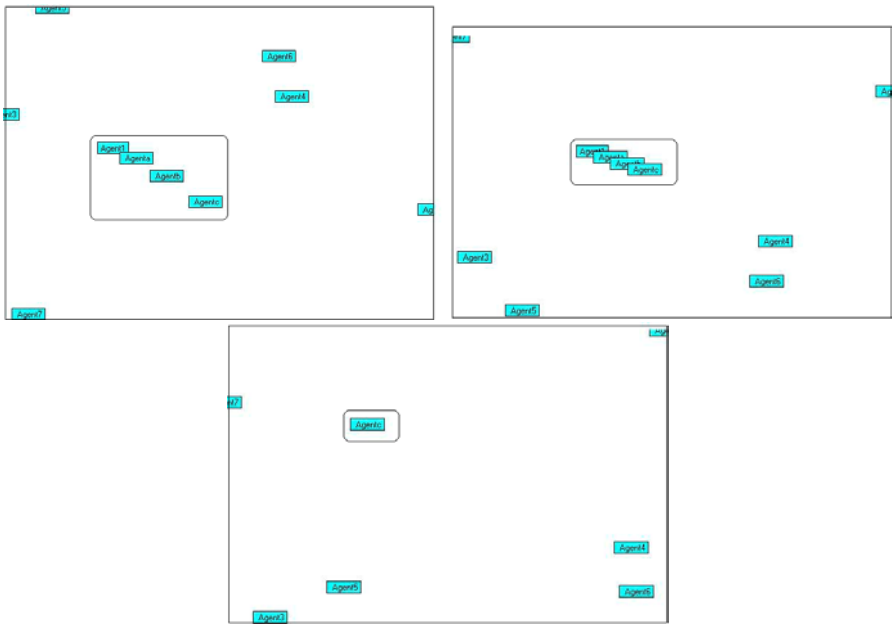

Fig. 1. City (top left), Euclidean (top right) and Angular Separation (bottom) measures followed by layout using Principal Coordinates Analysis. Agents a, b and c have scaled interest distribution of agent1. In the angular separation, agents a,b,c and 1 are located in the same position.

Step 1 - decide features of interest The first thing to consider is what features in the data one is interested in. We suppose that the aspects are: overlap of interest; intensity of interest; joint disinterest; similar pattern of interest (irrespective of subject). These elements provide a classification system with which we can construct a system to give numerical values to differences between a pair of agents' interests. Then these differences can be used to give a comparison measure for the behaviours of the various metrics. Statistical measures indicate the closeness of the match. The metrics may not correspond to the classification, even approximately. It could be that it is useful to use the classification system as the similarity measure itself and dispense with the metrics. However, in general, we are looking for a similarity metric that is not just a simple query, but something more subtle, something that reflects the multidimensional nature of the profiling data available. This corresponds to the scope that lies between the two questions:- Are you interested in sport? and Are you like me?

Also, if you are interested in sport, it may be valuable to know if you are a specialist or a generalist and in general terms what level your interest is on. Thus other similarity measures act as discriminators in this situation. Final choice of overall similarity measure may consist of additions of different similarity metrics (which may include results of specific queries) and can be arrived at in the manner of equation 5.

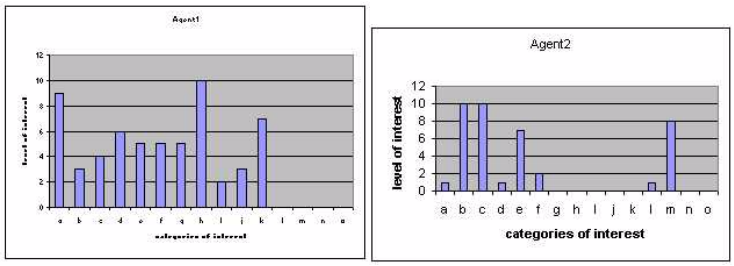

Fig. 2. Interests level against category for agents 1 and 2

The use of visualizations of data for pairs of agents can assist in the specification of features of interest. Simple diagrams such as bar and pie charts are helpful in designing a measure with which to make an informal assessment of the similarity between two agents. Figures 2, 3 are illustrative of this process and identify the features interest level and interest intensity that are used in step 2.

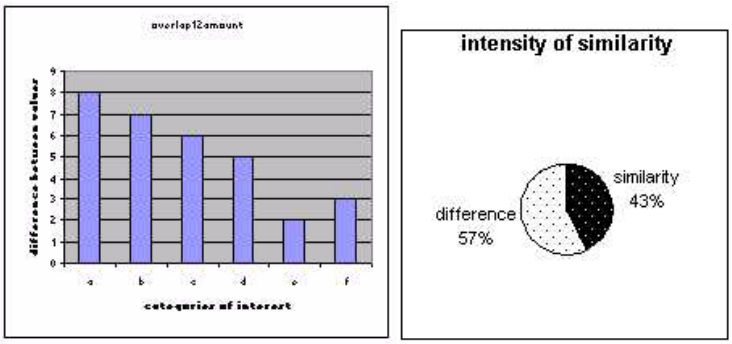

Fig. 3. Interests level difference against category for agents 1 and 2 (left) and intensity of overlap (right)

Step 2 - create a measure for the features to generate test data sets Suppose that types of agent similarity are chosen to examine e.g.: overlaps of three or more interests of high intensity; large overlaps irrespective of intensity with high common disinterest. Data is created for a representative member and edge member of desired clusters so that representative pairs of data can be created (or groups if required) to examine the metric behaviours both visually and by comparing distance matrices. To illustrate, a data set was created to produce examples covering the range of possibilities of overlap extent and intensity with respect to a reference agent's interests (as suggested by the visual

explorations of step 1). Then the metrics were examined to see how they clustered the group of similarities with number of overlap subjects $>= 3$ and intensity of overlap $>= 2/3$. One would expect the metrics to perform badly against this criterion, which is an example where a simple query would perform better (for instance, in the Yenta system, the matching between profiles is done simply on the basis of matching a single *granule*, which corresponds to a single interest, the metric is used in deriving the interest categories - if you simply want to exchange information on a subject that's ok, however such aspects as level of expertise are relevant, and finding like-minds needs greater subtlety). For metric discrimination, the distance comparison should be made by also evaluating the test criteria *for a number of other features* (such as joint disinterest and large overlaps irrespective of intensity) and combining the similarities.

Step 3 - evaluate visually and numerically The visual evaluation consists of visualizing the constructed data set and observing how well clustered the group of interest is. However, since the layout of such visualizations is an approximation (in order to satisfy the distances), and the observations not themselves measurements, evaluation by visualization is inexact. On the other hand, numerical evaluation, based on measuring differences between the estimated differences and the differences arrived at by the metric under consideration, is precise, but relies on the ability of the designer to define or estimate similarities between the data entities. For the example above this was done by awarding points according to number and intensity of topics of joint interest.

Figure 4 shows PCoA layout with City, Euclidean and Angular Separation differences, the reference agent is circled and the agents that are in my group of interest (according to the criteria in step 2) are indicated. That there is little difference between City and Euclidean indicates that it would be adequate to use city where time complexity was an issue. The three outlines traced by the points in the City and Euclidean plots correspond closely to the classification system and the group of interest is well clustered in visual terms. The Angular Separation plot does not cluster so well, misplacing three agents. The layout of the angular separation distances is actually a screenshot of a 3D representation as the layout was particularly inaccurate and needed the extra dimension to improve it (the first two eigenvalues accounted for only 38% of the variance in the data and the first three for only 48%). The inaccuracy of this layout highlights the difficulty of using visualization to assess similarity.

5 The Use of Position Instead of Vector for Profile

The pictures of information spaces as maps or terrains derived from multivariate data using self-organizing maps [16] provide us with a compelling image of the profile or topic space we are exploring. The metrics discussed above generate similar conceptual spaces when visualized. Yet this is a misleading image, since the data are high dimensional and it is impossible to represent their similarities accurately in 2 or 3D space (direct mapping methods for multivariate data, such as colour maps and parallel coordinate plots, are not included in this comment). Nevertheless, as an approximation and as a representation, an overview perhaps, of a large body of entities, it is being found useful (see e.g. [20]). Suppose we assume the validity of the layout and propose that

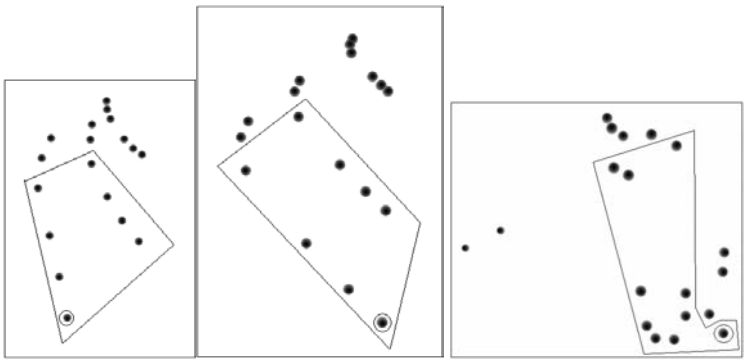

Fig. 4. Constructed data set with (left to right) PCoA and City, Euclidean and Angular differences.

the agent carries with them their xy (or xyz) coordinates and uses them as their profile. When meeting a fellow agent they can ask for the agent's xy coordinates and compute the Euclidean distance to calculate their similarity. This would be more efficient than carrying a potentially long profile vector and enable them to use their profile without revealing details or requiring encryption. Two different ways of using this idea suggest themselves - calculating back at base and on the fly.

5.1 By base calculation

The agents both have the calculations done at a base point and periodically return for updates. Here the error will be that of the layout itself and the agent would be able to have details of the mean error and variance supplied with its coordinates, so that it can take this into account. Figure 5 shows the layout after City distance and PCoA of the seven agents of randomly generated data from above. Thus, if Agent1 meets Agent2 they can compare coordinates, ((-12.30, -5.20),(23.27,-8.44)), to calculate the Euclidean distance to give them the distance they are apart in this map.

5.2 By calculation on the fly

Here the agent calculates its position with respect to a number of reference vectors (either dynamically or at an earlier point in time) and then compares with another agent's position calculated similarly. Using the seven agent random data again, the reference vectors are chosen to be agents 5,6 and 7 illustrated in the generic data section. Three reference agents are the minimum since only two will create two possible arrangements when agents 1 and 2 overlay their positions. Agents 1 and 2 separately calculate their City distances to the three reference vectors and subsequently lay out these distances with PCoA as shown in figure 6.

They now have xy coordinates, but in order to compare them they must be scaled (the Euclidean distance between 5 and 6 is used here), centered (here Agent 5 is placed

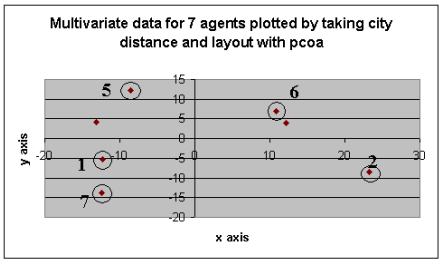

Fig. 5. Illustration of base plot, the three reference agents (5,6 and 7) and the two of interest in this measurement (1 and 2) are circled.

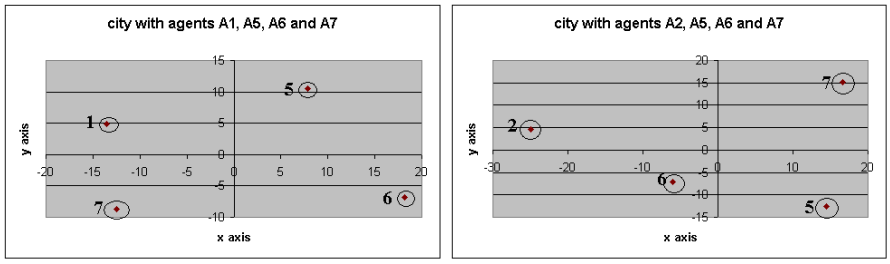

Fig. 6. Illustration of plots calculated individually by agents 1 (left) and 2 (right) with respect to the three reference agents (5,6 and 7) as circled and numbered.

at 0,0) and finally rotated to bring the agents 5,6 and 7 into position. Now the coordinates of the agent's position are in a form that they can use for comparisons. The results of the base calculation and on-the-fly calculation of the difference between agents 1 and 2 are given in the table below. (Since these are normalized with respect to the distance between agents 5 and 6, a value of 1 would indicate that they were the same distance away from each other as agents 5 and 6 are)

original city dist	base dist	on-the-fly dist
1.64	1.77	1.57
exact	8%err	-4%err

6 Conclusions and Future Work

Visual datamining seeks to increase the integration of visualization with specific datamining techniques. This paper presents two applications with this in mind.

Appropriate clusterings of data are sought, whilst at the same time layouts are required to present overviews. The user needs understanding of the layout algorithm to appreciate the implications of the overview, the implication of arriving at different clusterings with different algorithms needs to be understood by those seeking valid cluster-

ings and classifications. These two purposes concern the same process, but are subtly different in their focus. The first application described in this paper, illustrating the use of signature exploration in making the behaviour of the similarity metrics more concrete and assisting in similarity metric choice, is an example of the merging of understanding of overview and determining appropriate metric. Visualization of pairs of data helped in the creation of an ad hoc, user-specific, classification with which to assess the overview and thus also the metrics. An obvious next step is to use feedback to select and modify metrics and features and this is another part of the signature exploration process. Continuing work lies in further developing the interface for exploration, the data construction engine and in conducting usability tests.

Visualizations sometimes suggest the idea of a topic or similarity space - looking at a 2D or 3D scatterplot the closeness of entities is intuitively understood as similarity. Where dimension reduction is involved, considerable approximation or abstraction is required. If this is a valid procedure (in the sense of the considerable error sometimes incurred), and such diagrams are widely used without warnings given, then the idea of using location as a form of privacy protection (the transformation is a one-way function) must also hold on some level. The simple example for using position as a profile demonstrated in this paper - a potentially most useful mechanism - is encouraging, now evaluation for many different data sets is required to test its robustness, in terms of whether the original profile is fully protected and the tolerance of approximation in *locations* of the entities.

Evaluation of the position-as-profile concept points to one of our most pressing problems in visualization - how valid are our visualizations when dealing with complex data and involving approximation or abstraction? How can the level of approximation be indicated to the viewer? Correspondingly, how can a measure of confidence in the agent's location in the interest space be given to the agent? The investigation of the position-as-profile idea is the same investigation as that of the validity of layout. Thus, we begin to think in terms of transfering our picture as a viewer to the agent, so that the two can become one - a kind of viewer/agent entity. The agent thus may be a software agent or a human agent. The question now becomes, how can the boundaries or parameters of the validity be described to the viewer/agent? How can they be encoded visually and in software terms? We interchange the viewer with agent and must express what the user *sees* (or finds useful) in a form that the software agent can work with. Via the agent paradigm we may thus be helped toward creating programs that can use graphical elements to mimic our visual thinking.

6.1 Acknowledgements

This work is supported by the EPSRC and British Telecom (CASE studentship - award number 99803052).

References

1. S. K. Card, J. D. Mackinlay, and B. Shneiderman. *Readings in Information Visualization: Using Vision To Think*. Morgan Kaufmann, 1999.
2. C. Chen. *Information Visualisation and Virtual Environments*. Springer, 1999.
3. G. di Battista, P. Eades, R. Tamassia, and I. G.Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall, 1999.
4. L. Foner. Yenta: a multi-agent, referral-based matchmaking system. In *The First International Conference on Autonomous Agents, Marina del Rey, California*. ACM press, 1997.
5. L. N. Foner. Clustering and information sharing in an ecology of cooperating agents. In *AAAI Spring Symposium '95 on Information Gathering in Distributed, Heterogeneous Environments, Palo Alto*, 1995.
6. J. Glasgow, N. Hari Narayanan, and B. Chandrasekaran. *Diagrammatic Reasoning*. AAAI Press / The MIT Press, 1995.
7. A. D. Gordon. *Classification*. Chapman and Hall / CRC, 2nd edition edition, 1999.
8. P. Noy and M. Schroeder. Introducing signature exploration: a means to aid the comprehension and choice of visualization algorithms. In *ECML-PKDD01 Visual Data Mining Workshop*, Freiburg, Germany, Sept 2001.
9. E. Ogston and S. Vassiliadis. Matchmaking among minimal agents without a facilitator. In *Proceedings of Autonomous Agents2001*, Montreal, Canada, 2001. ACM press.
10. P.Faratin, C.Sierra, and N.R.Jennings. Using similarity criteria to make negotiation tradeoffs. In *Proc. of 4th Int. Conf. on Multi-Agent Systems ICMAS-2000*, pages 119–126, Boston, USA, 2000. IEEE Computer Society.
11. M. Schroeder. Using singular value decomposition to visualise relations within multi-agent systems. In *Proceedings of the third Conference on Autonomous Agents*, Seattle, USA, 1999. ACM Press.
12. M. Schroeder, D. Gilbert, J. van Helden, and P. Noy. Approaches to visualisation in bioinformatics: from dendrograms to Space Explorer. *Information Sciences*, 139:19–57, 2001.
13. M. Schroeder and P. Noy. Multi-agent visualization based on multivariate data. In *Proceedings of Autonomous Agents2001*, Montreal, Canada, 2001. ACM press.
14. G. Somlo and A. Howe. Incremental clustering for profile maintenance in information gathering web agents. In *Proceedings of Autonomous Agents2001*, Montreal, Canada, 2001. ACM press.
15. R. Spence. *Information Visualization*. Addison-Wesley, 2001.
16. T.Kohonen. *Self-organising maps*. Springer-Verlag, 2nd edition edition, 1997.
17. D. Veit. *Matchmaking algorithms for autonomous agent systems*. Master's thesis, Institute of Computer Science, University of Giessen, Germany, 1999.
18. D. Veit, J. Muller, M. Schneider, and B. Fiehn. Matchmaking for autonomous agents in electronic marketplaces. In *Proceedings of Autonomous Agents2001*, Montreal, Canada, 2001. ACM press.
19. A. Webb. *Statistical Pattern Recognition*. Arnold, 1999.
20. Websom. http://websom.hut.fi/websom/.
21. G. Weiss, editor. *Multiagent Systems*. MIT Press, 1999.