# *Genome-based Population Clustering: Nuggets of Truth Buried in a Pile of Numbers?*

Marina Ioannou[1], George P. Patrinos[2], Giannis Tzimas[3]

[1] Department of Computer Engineering and Informatics, Faculty of Engineering, University of Patras, Patras, Greece
ioannouz@ceid.upatras.gr
[2] Department of Pharmacy, School of Health Sciences, University of Patras, Patras, Greece
gpatrinos@upatras.gr
[3] Department of Applied Informatics in Management & Economy, Faculty of Management and Economics, Technological Educational Institute of Messolonghi, Messolonghi, Greece
tzimas@teimes.gr

**Abstract.** National/Ethnic population Mutation databases (NEMDBs) are online mutation depositories recording extensive information about the described genetic heterogeneity in populations and ethnic groups worldwide. FINDbase (http://www.findbase.org) is a database containing causative mutations and pharmacogenomic markers allele frequencies in various populations and ethnic groups. In this paper, we experiment with designing and applying new automated data mining techniques on the original FINDbase causative mutations data collection in an attempt to identify genomic relationships between populations. Furthermore, we have developed an interactive web-based visualization tool that enables users to correlate the information, determine the relationships and gain insight into the underlying data collection in a novel and meaningful way.

**Keywords:** Data Mining, Genome-based Population Clustering, Data Visualization.

## 1 Introduction

In the last decades, Genetic and Mutation databases concerning human genes have become increasingly important in many fields of biology and medicine. The completion of the human genome sequence, as well as the development of new methods for the determination of point mutations, have recently led to a tremendous increase of mutations identification, in a growing number of genes. Consequently, it is obvious that the organization and management of these alteration data is of great importance to the scientists and researchers in genetic and genomic research.

Mutation databases are large mutation data repositories that provide valuable genomic insights and genotype-phenotype correlations pertaining to monogenic inherited disorders. Currently, the complex research area of Disease Genetics involves three main types of genetic databases, general or core mutation databases (GMDs), locus-


© Springer-Verlag Berlin Heidelberg 2011

specific databases (LSDBs), and National/Ethnic Mutation databases (NEMDBs). GMDs are databases that collect all described mutations in all genes; LSDBs are repositories for just one or few genes usually related to a single gene disorder, while NEMDBs describe the genetic composition and the allele frequencies of a population or ethnic group. Among the most prevalent examples of mutation databases are the Human Gene Mutation Database (HGMD) [1] and Online Mendelian Inheritance in Man (OMIM) [2].

Although considerable progress has been made recently in this area, dealing with genomic information still remains a difficult and challenging task, as data are heterogeneous, huge in quantity and geographically distributed. As a result, there is an urgent need for developing advanced and efficient computational methods and tools to facilitate the process of managing and discovering useful hidden patterns and knowledge from these large and complex biomedical data repositories.

In this work, we experiment with data acquired from FINDbase [3-5]. FINDbase is an online resource (http://www.findbase.org) documenting frequencies of pathogenic genetic variations leading to inherited disorders in various populations worldwide. We are envisaging and applying efficient data mining techniques on the existing FINDbase population genomic data collection aiming to explore meaningful and useful hidden genetic correlations among populations and their dependency on the geographic distance of populations. In addition, we experiment with combining an automated data mining process with novel visualization techniques. The utilization of both automatic analysis methods and human perception is a promising process for more effective inspecting, understanding and interacting with these huge and complex data collections.

The rest of this paper is organized as follows. Section 2 provides a detailed description of the proposed data mining process, Section 3 presents the web-based visualization software; Section 4 analyzes the results of the clustering method and finally Section 5 points out our conclusions along with ideas for future work regarding this area.

## 2 Data Mining Process

The data mining process proposed in our work aims at the identification of population clusters based on their mutation's allele frequency similarity and further sub-clusters regarding their geographic location similarity.

In particular, we have designed a data clustering algorithm that involves two levels of clustering and is based on a combination of different techniques, such as hierarchical clustering and spherical k-means algorithm [6].

The first level of the whole process aims to identify population clusters based on their mutation’s allele frequency similarity. As a first step, we preprocess the initial FINDbase dataset in order to convert the underlying data to a more processable structure. Particularly, we represent each population as a vector based on its relative mutations. Subsequently, applying Latent Semantic Indexing (LSI) and Coarse Clustering we generate an initial partition of populations. In the case that the number of the ini-

tial clusters is relatively large, we reduce the number of these clusters by applying an agglomerative Hierarchical Clustering algorithm. Finally, we refine the result of clustering utilizing the Spherical K-Means Algorithm.

The main goal of the second level of our data mining process is to generate sub-clusters within the clusters already identified in the first step. Firstly, we represent each population using a spatial representation according to the geographical coordinates as they exist in the original FINDbase dataset. Then, for each cluster we apply an agglomerative hierarchical algorithm in order to create sub-clusters containing populations with similar geographic location.

The basic steps of the whole process are depicted in Fig. 1 and in the next section we describe these steps in detail.

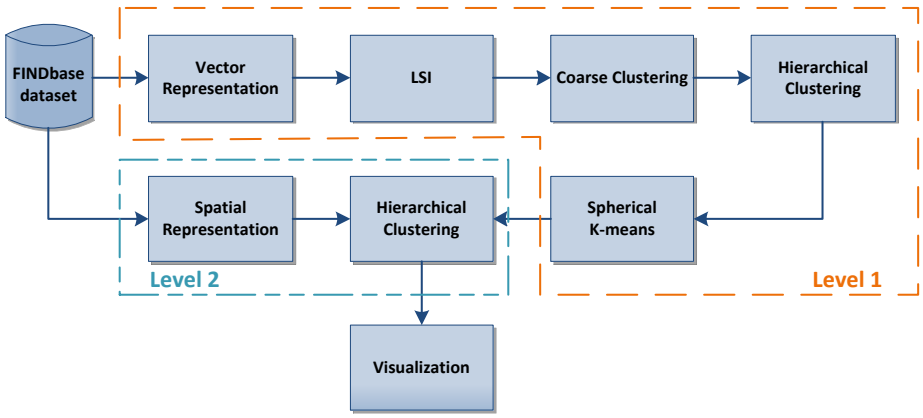


**Fig. 1.** Data Mining Process

## 2.1 The Algorithm in Detail

**Vector and Spatial Representation.**
The vector space model is one of the most popular in ad-hoc information retrieval, bearing many uses in various kinds of operations. According to the vector space model each object is represented as a vector of features.

In our application, we utilize the vector space model in order to represent each population as a feature vector (1).

$$d = \{f_1, f_2, \dots, f_m\} \tag{1}$$

The vector d represents the causative mutations of a specific population. More specifically, d is an m-dimensional vector, where m is the number of unique causative mutations. For a specific population the contribution of each mutation in its content representation is the genetic variation allele frequency, i.e. the number of times the allele of interest is observed in this population divided by the total number of copies

of all the alleles at that particular genetic locus in this population. The vector d takes part in the first level of population clustering, i.e. in clustering based on mutation similarity.

According to the Spatial Representation, each population is also represented by a single point in the geographic space (2).

$$P = (x, y) \tag{2}$$

The point P represents the Geographic coordinates of a specific population. Particularly, P is a point in the two dimensional geographic space where x and y are the geographic latitude and longitude respectively and takes part in the second level of population clustering, i.e. in the clustering based on geographic location similarity.

Finally, as a product of the vector representation, a matrix A is produced of $m \times n$ dimension, where m is the number of unique causative mutations and n is the number of unique populations of our data set.

**Latent Semantic Indexing.**

The Latent Semantic Indexing (LSI) [7] is an indexing and retrieval method that uses the Singular Value Decomposition (SVD) technique, in order to identify patterns in the relationships between the terms and concepts contained in an unstructured collection of text.

The successful performance of LSI in textual information retrieval has led to its application on numerical data too. In this case, the Latent Semantic Indexing is performed in order to capture the underlying structure of data associations and to discover hidden patterns in a more efficient way. The SVD decomposes the $m \times n$ matrix $A$ into a product of three simpler matrices (3),

$$A=USV^T \tag{3}$$

where $S$ is a diagonal matrix which contains the singular values of $A$. In our method, we keep only the $k$ singular values, according to a prespecified threshold, and we produce the matrices $U_k,S_k,V_k$. Finally, we normalize the columns of matrix $A$, so that all populations' vectors turn into unit vectors.

**Coarse (initial) Clustering.**

Fuzzy Clustering [8] has been widely used in the area of information retrieval and data mining. In traditional Fuzzy Clustering methods data elements can belong to more than one cluster, and each data element is associated to every cluster based on a membership function, that differentiates on the specific cluster. In our approach, we interpret the results of the LSI transformation of the initial mutation-population matrix as a form of fuzzy clustering, and then we assign each population to exactly one cluster, the cluster where the population has the highest degree of participation, thus producing an initial rough clustering. In particular, based on the $n$x$k$ matrix $V_kS_k$, produced by SVD, we interpret the $k$ columns of the $V_kS_k$ matrix as a set of $k$ clusters and the $n$ rows as the populations. Each element $(i, j)$ of the $V_kS_k$ matrix, where $i$ is the

row and *j* is the column, defines the population's *i* degree of participation to the cluster *j*. This clustering is transformed to a crisp clustering by assigning each population to the cluster, where the population has the highest degree of participation according to the $V_k S_k$ matrix.

Since the produced clustering, is based on an heuristic interpretation of the LSI machinery, we consider it to be an initial grouping, that is used as the preprocessing phase that feeds the remaining components of the clustering phase. In particular note, that the produced number of clusters is equal to the LSI dimension which is generally high, hence the clusters need to be reduced (this is accomplished with a hierarchical algorithm) and redefined (this takes place with calls of the spherical k-means procedure). Finally, following similar representations (see e.g. [9]), for each cluster we calculate a centroid *c* which is nothing more than the vector obtained by averaging the weights of the various mutations present in the populations of this cluster.

**Hierarchical Clustering.**

An agglomerative hierarchical clustering algorithm (see [9]) is applied in both the levels of our clustering approach.

In the first level of clustering, we apply an agglomerative hierarchical clustering algorithm (see [9]) in order to reduce the number of clusters produced by Fuzzy Clustering, from k to K, where K is the desired number of final clusters. K is also the initial number of clusters, which is given as an input to the spherical k-means algorithm in the next step. Based on the UPGMA similarity measure (see [9]), the hierarchical clustering algorithm merges the two most similar clusters at each step, and we stop its functioning when reaching a value of K clusters.

In the second level of the clustering method, we also apply this agglomerative hierarchical clustering algorithm in order to identify further sub-clusters of population clusters generated from the Spherical K-means algorithm, regarding their geographic location similarity. In particular, we use the spatial representation of populations, as we mentioned above. At each step, the hierarchical clustering algorithm merges the two most similar clusters according to the Euclidean distance.

**Spherical K-Means Algorithm.**

The spherical *k*-means algorithm is a well studied variant [10] of the *K*-means Algorithm with cosine similarity as the selected distance metric, and it is well known for its efficiency in clustering large data collections. One of the main disadvantages of spherical *k*-means is the fact that the initial clusters and centroids are randomly selected. For this reason, in our case we define as initial clusters the clusters generated from the Hierarchical Algorithm of the first level of clustering, and we utilize the "ping-pong" refinement algorithm [10-11] which consists of two steps: spherical *k*-means and Kernighan-Lin heuristic.

### 2.2 Technologies used

The whole mining process has been implemented in C Sharp (C#) utilizing Microsoft Visual Studio 2010. The software receives as input the original FINDbase dataset and outputs a set of text files that contain the results of the clustering algorithm. Subsequently, this set of text files is fed as input to the visualization software. In order to represent vectors, matrixes and to calculate Singular Value Decomposition and other arithmetic operations between vectors and matrixes, we utilize the SmartMathLibrary [12] which provides scientific computing to the .NET platform.

## 3 Visualizing the Results

FINDbase Genome-based Population Clustering is a web-based tool that lets users to explore genetic relationships between various populations through an interactive visualization. It represents the results of our clustering method in a more effective and understandable way. Furthermore, it allows users to view additional information for a specific population, such as causative mutations and their allele frequencies, as well as its geographic location through a map. The visualization environment consists of the main clustering visualization, two data tables and a map (Fig.2). You can have access to the FINDbase Genome-based Population Clustering environment at: http://www.biodata.gr/findbase/GenomebasedPopulationClustering/.

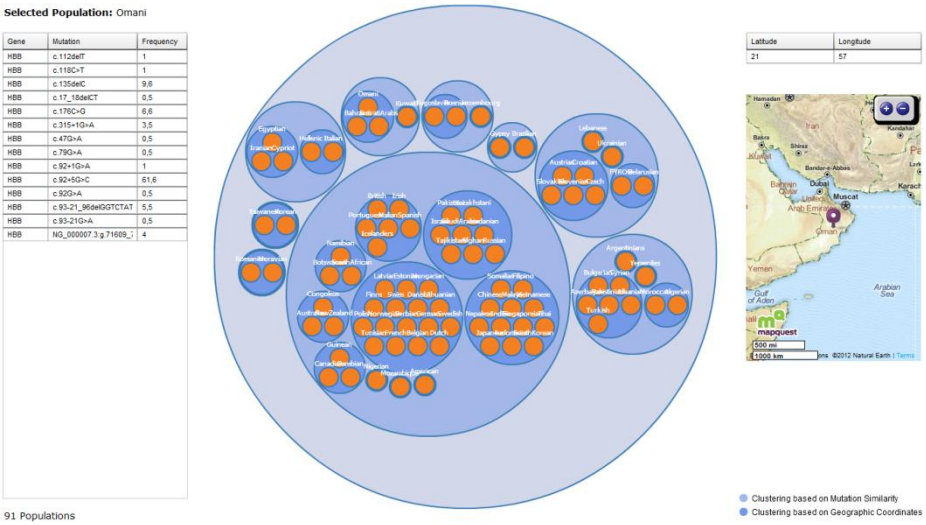


**Fig. 2.** The clustering visualization environment

The main part of the Genome-based Population Clustering visualization is based on a circle packing layout. Circle packing layout places nodes as circles compacted into a larger circle. The brother nodes at the same level are represented by tangent circles and all children of a node are packed into that node in order to visualize the

hierarchy. The algorithm of circle packing layout [13] has the main advantage that is effective for visualization of large amounts of hierarchical structured data.

According to the circle packing layout, the main part of the visualization represents the two levels of population clustering. The color saturation of nodes indicates the level of clustering (the more saturated, the greater the level of clustering). The leaf nodes (orange color) represent the populations and the labels above the leaf nodes indicate the population name.

At the left area of the visualization environment, a detailed data grid is positioned that shows the genes, the causative mutations and their allele frequencies of the selected population. At the right area, a detailed data grid and a map are positioned that present the values of geographic coordinates and the geographic location of the selected population respectively.

By clicking on a population node, the user has the ability of viewing further details for the selected population. Specifically, the left data grid shows all the causative mutations and the allele frequencies that are present in the selected population, while the right data grid and the map present the geographic location. In addition, when clicking on a cluster node, it expands or collapses this node.

When hovering the mouse over a node additional information is displayed, such as the population name in the case of a leaf node.

### 3.1 Technologies used

The Genome-based Population Clustering has been implemented as a web Flash application using the Adobe Flex platform and the Flex 4.6 software development kit, in order to achieve a high level of compatibility with a wide variety of operating systems and browsers. The underlying data structures and functionalities have been implemented in ActionScript 3 and the Flex component library has been used in order to exploit its advanced capabilities. Furthermore, we have utilized the Open Flash Maps API version 7.1.1_OSM [14] for representing the geographic location of the selected population. Our decision to use Flash as our visualization platform was due to the fact that it is nearly ubiquitous on a variety of internet-enabled desktop as well as mobile operating systems.

The main visualization area of the Genome-based Population Clustering has been developed using the Flare Visualization Toolkit [15]. Flare is an open-source library written in ActionScript 3 [16], an object-oriented programming language, for creating data visualizations that run in the Adobe Flash Player. Including a wide variety of features, ranging from basic charts to complex interactive graphs, it supports data management, visual encoding, animation, and interaction techniques. Flare is a Flash version of its predecessor Prefuse [17], a visualization toolkit for Java. Instead of creating Java applications with complex visualizations, Flare offers the ability of developing thin-client, web-based and rich interactive experience environments. It has already been used in many well-known web-based visualization applications such as the Many-Eyes website [18], build by the IBM Visual Communication Lab, for user-contributed data visualization as well as the BBC SuperPower website [19] for mapping the top 100 sites on the Internet.

## 4 Results

The data mining process produced 9 clusters of populations according to their mutation's allele frequency similarity (Fig. 3). Analyzing these clusters, we observe some interesting relationships between populations. More precisely:

**Cluster A.** The populations of Cluster A present mainly genetic causative mutations in the gene *HBB*. The Omani and Emirati Arabs populations are among the most representative examples of this cluster. These two populations have 9 *HBB* gene mutations in common and the average difference of their common mutations' allele frequencies is about 3.5%.

**Cluster B.** In Cluster B, the populations present mainly genetic causative mutations in the *HBB* and *CFTR* genes. The Syrian and Palestinian populations are among the most representative examples of this cluster. These two populations have in common 7 mutations in the *HBB* gene and the average difference of their common mutations' allele frequencies is about 4.29%.

**Cluster A and Cluster B**. Comparing the populations of Cluster A and B, we observe that they have some mutations in the same genes in common. However, the populations of these clusters present large differences in the allele frequencies of their common mutations. For example, the Palestinian and Omani populations have in common 5 mutations in the gene *HBB*. Nevertheless, the average difference of their common mutations' allele frequencies is about 19%.

**Cluster C**. Cluster C populations present mainly genetic causative mutations in the *CFTR* gene. The Slovenian and Austrian populations are among the most representative examples of this cluster. These two populations have 8 *CFTR* gene mutations in common and the average difference of their common mutations' allele frequencies is about 2.4%.

**Cluster C and Cluster B.** Comparing the populations of Cluster C and B, we observe that these populations have some common mutations in the *CFTR* gene. Nevertheless, they present large differences in the allele frequencies of their common mutations. For example, the mutation *CFTR*: p.508delF is present in both Slovenian and Turkish populations. However, the allele frequency of *CFTR*: p.508delF in the Slovenian population is 37.91%, while in Turkish population is only 2.45%.

**Cluster D**. The populations of Cluster D present common genetic causative mutations in the *PAH* gene. In particular, the Romanian and Moravian populations have 5 *PAH* gene mutations in common and the average difference of their common mutations' allele frequencies is about 3.7%.

**Cluster E.** In Cluster E, the populations present mainly common genetic causative mutations in the *HBB*, *CFTR*, *PAH*, *HBA* and *SERPINA1* genes. For example, the Hellenic and Italian populations have in common 57 mutations in the genes *HBB*, *CFTR*, *PAH*, *HBA* and *SERPINA1* and the average difference of their common mutations' allele frequencies is about 1.98%.

In summary, the above analysis examples of results arisen from the clustering method indicate that there are various hidden meaningful genetic correlations among populations that our technique successfully captures and reveals.

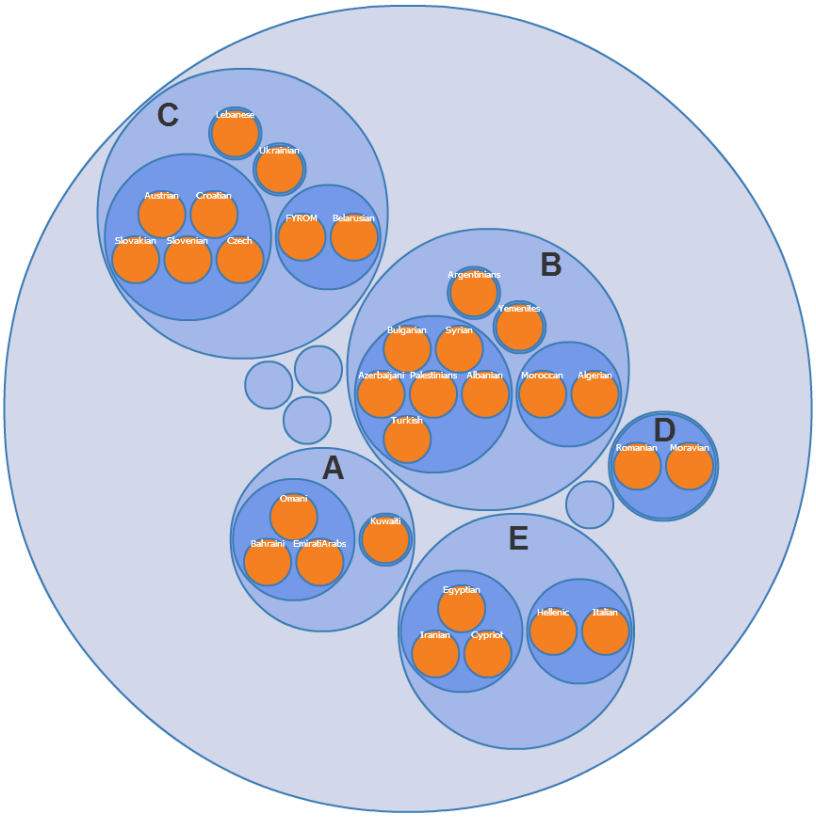


**Fig. 3.** A first level visualization of the results

## 5 Conclusions and Future Work

In this work, we present a first attempt at integrating novel clustering techniques in the FINDbase genomic population data set for grouping populations into clusters according to their genomic similarities aiming to reveal hidden meaningful correlations among populations. Moreover, we developed a web-based visualization tool, namely, the Genome-based Population Clustering that supports interactivity, animation and consequently it displays the results of clustering method in a more understanding way.

Our future plans include further experimenting with applying more alternative data mining analysis techniques. In addition, we are going to combine the two different representations of populations in a more sophisticated way aiming to improve the effectiveness of the clustering method.

Our primary goal will always be to provide a powerful and stable tool that supports high-level multidisciplinary knowledge and it is open, available and visible to all members of the worldwide scientific community.

## 6 References


1. Human Gene Mutation Database (HGMD), http://www.hgmd.cf.ac.uk/ac/index.php
2. Online Mendelian Inheritance in Man (OMIM), http://www.ncbi.nlm.nih.gov/omim
3. Van Baal, S., Kaimakis, P., Phommarinh, M., Koumbi, D., Cuppens, H., Riccardino, F., Macek, M. Jr., Scriver, C.R., Patrinos, G.P.: FINDbase: A relational database recording frequencies of genetic defects leading to inherited disorders worldwide. Nucleic Acids Res., 35(Database issue), D690-D695 (2007)
4. Georgitsi, M., Viennas, E., Gkantouna, V., Christodoulopoulou, E., Zagoriti, Z., Tafrali, C., Ntellos, F., Giannakopoulou, O., Boulakou, A., Vlahopoulou, P., Kyriacou, E., Tsaknakis, J., Tsakalidis, A., Poulas, K., Tzimas, G., Patrinos, G.P.: Population-specific documentation of pharmacogenomic markers and their allelic frequencies in FINDbase. Pharmacogenomics 12(1), 49-58 (2011)
5. Georgitsi, M., Viennas, E., Gkantouna, V., van Baal, S., Petricoin, E.F., Poulas, K., Tzimas, G., Patrinos, G.P.: FINDbase: A worldwide database for genetic variation allele frequencies updated. Nucleic Acids Res. 39(Database issue), D926-D932 (2011)
6. Ioannou, M., Makris, C., Tzimas, G., Viennas, E.: A Text Mining Approach for Biomedical Documents, In: 6th Conference of the Hellenic Society for Computational Biology and Bioinformatics (HSCBB11), Patras, Greece (2011)
7. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval: the concepts and technology behind search, ACM Pres Books (2010)
8. Inoue, K., Urahama, K.: Fuzzy Clustering based on Cooccurence Matrix and Its Application to Data Retrieval. Electron. Comm. Jpn. Pt. 2 84,10-18 (2001)
9. Steinbach, M., Karypis, G., Kumar, V.: A Comparison of Document Clustering Techniques, In: KDD Workshop on Text Mining (2000)
10. Kogan, J.: Introduction to Clustering Large and High-Dimensional Data, pp. 51-72. Cambridge University Press (2007)
11. Dhillon, I.S., Guan, Y., Kogan, J.: Iterative Clustering of High Dimensional Text Data Augmented by Local Search, In: 2nd IEEE International Conference Data Mining, ICDM (2002)
12. SmartMathLibrary, http://smartmathlibrary.codeplex.com/
13. Wang, W., Wang, H., Dai, G., Wang, H.: Visualization of Large Hierarchical Data by Circle Packing, In: CHI (2006)
14. MapQuest, http://developer.mapquest.com/
15. Flare - Data Visualization For The Web, http://flare.prefuse.org/
16. ActionScipt 3, http://www.adobe.com/devnet/actionscript.html
17. Prefuse visualization toolkit, http://prefuse.org/
18. Many-Eyes, http://www-958.ibm.com/software/data/cognos/manyeyes/
19. BBC SuperPower, http://news.bbc.co.uk/2/hi/technology/8562801.stm