Visual Representation for Data Clustering

FINAL PROJECT DESCRIPTION

Jorge Henrique Piazentin Ono
jpo286@nyu.edu - jpo286
Paola Tatiana Llerena Valdivia
paoyo1@gmail.com

Project page: https://github.com/nyu-cs6313-fall2015/Group-12/
Video: https://vimeo.com/149582643
Working demo: http://nyu-cs6313-fall2015.github.io/Group-12/

1. What is the problem you want to solve and who has this problem?

Clustering is the process of grouping similar objects together (Murphy, 2012). It is a fundamental data analysis tool that has been used in many applications. However, understanding the result of a clustering algorithm is a challenging task and data analysts face this problem daily (Cao, 2011). Visual aids have been developed to help solve this problem.

Many visualization techniques were proposed to explore clustering results, but most of them are not suitable to handle categorical data. A technique that efficiently deals with this kind of attributes would help the analysis of many real world problems, for example, medical records, containing disease, type of treatment, cost, among many other entries.

The goal of this project is to develop visualization techniques that enable users to understand the resulting clusters and their semantics. We will focus on the analysis of hierarchical clustering, as it can be easily generalised to deal with categorical data. More specifically, we want to understand why each cluster was formed (for example, which dimensions contributed the most) and, conversely, why an item belongs to a cluster. Our technique should be sufficiently generic to handle data of different natures, such as categorical and quantitative. For this reason, we have chosen a medical data set to steer and evaluate the proposed methodology.

The medical data set will be described later in this text. Users that could be interested in understanding this data include health care providers, governments and health insurance companies, which could use the information gathered to improve their services.

2. What questions do you want to be able to answer with your visualization?

In terms of general cluster analysis our visualization should be able to answer the following questions:

- Why was a cluster formed by the algorithm?
  - Is it because there is a group of dimensions with similar values within the cluster?
  - Is it because their elements are very different from the rest of the data set?
- What is a good number of clusters for a specific data set?
  - Is there a way to choose the number of clusters based on a visualization?
- Why does an item belongs to a cluster?
  - How do the entries of this item compare to the entries of other items in the cluster.

In terms of the medical data set:

- Why are a set of patients grouped together by the algorithm?
  - How are the grouped patients similar to each other? Do they share the same disease, treatment, age, etc?
- How many types of patients are there in the population?
  - What are their main traits?
- Should the government / health care provider / insurance company create new policies for the discovered patient groups?
  - Can visualization improve the quality of the treatment provided?

3. What is your data about? Where does it come from? What attributes are you going to use? What is their meaning? What are their attribute types (data abstraction)? Do you plan to generate derived attributes? If yes, which and why?

We are going to analyse data related to medical information of real world patients. which comes from Centers for Medicare and Medicaid Services - CMS (https://www.cms.gov/Research-Statistics-Data-and-Systems/Downloadable-Public-Use-Files/SynPUFs/DE_Syn_PUF.html).
The data set was collected in the period of 2008 - 2010 and contains 116353 records. In order to enable the analysis of the data in the browser, we randomly sampled 15000 records of the data.

More than three thousand attributes are available in the spreadsheet. Here we describe the most relevant among them. The attribute Cluster ID is derived from the original data and represents the cluster each instance belongs to.

Cluster ID is computed based on an hierarchical clustering algorithm. Firstly, the data is pre-processed (each quantitative bin is normalized between 0 and 1). Then, we calculate a distance matrix of the data with the Gower coefficient (Gower, 1971). Finally, the hierarchical clustering is computed using the complete linkage algorithm.

Attribute Name	Type	Description	Possible values
age	Quantitative	Age of patient	[28,102]
claim cost	Quantitative	Cost of treatment	[$-7290, $252010]
county	Categorical	County code of patient	{0, 1, ... 999}
state	Categorical	State of patient	52 states
dead	Binary	Status of patient - dead	{Yes, No}
gender	Categorical	Gender of patient (Male, Female, Not available)	{M, F, N/A}
num claims	Quantitative	Number of claims from patient	[0, 289]
race	Categorical	Race of patient (black, hispanic,	{b, h, o, w}

		oriental, white)	
chronic alzheimer	Binary	Patient has chronic alzheimer	{Yes, No}
chronic arthritis	Binary	Patient has chronic arthritis	{Yes, No}
chronic cancer	Binary	Patient has chronic cancer	{Yes, No}
chronic depression	Binary	Patient has chronic depression	{Yes, No}
chronic diabetes	Binary	Patient has chronic diabetes	{Yes, No}
chronic heart failure	Binary	Patient has chronic heart failure	{Yes, No}
chronic ischemic heart disease	Binary	Patient has chronic heart disease	{Yes, No}
chronic kidney disease	Binary	Patient has chronic kidney disease	{Yes, No}
chronic obstructive pulmonary disease	Binary	Patient has chronic obstructive pulmonary disorder	{Yes, No}
chronic osteoporosis	Binary	Patient has chronic osteoporosis	{Yes, No}
chronic stroke	Binary	Patient has chronic stroke	{Yes, No}
diagnosis [code of diagnosis]	Binary	Patient was diagnosed with [disease]	{Yes, No}
procedure [code of procedure]	Binary	Patient was treated with procedure [procedure]	{Yes, No}
Cluster ID (**derived attribute**)	Categorical	Cluster the record belong to	{1, 2, 3, … }

4. What have others done to solve this or related problems?

Clustering is the process of grouping similar objects together into subsets called clusters, such that data entities in each cluster are similar in some way (Murphy, 2012), (Cao, 2011). There are two types of clustering algorithms available in the literature: partitional clustering and hierarchical clustering. In partitional clustering, the objects are divided into disjoint sets. In hierarchical clustering, a nested tree of partitions is created and the user can navigate in that hierarchy (Murphy, 2012). Figure 1 shows a taxonomy of clustering algorithms. A thorough review of clustering algorithms can be found in the survey of Anil et al. (1999).

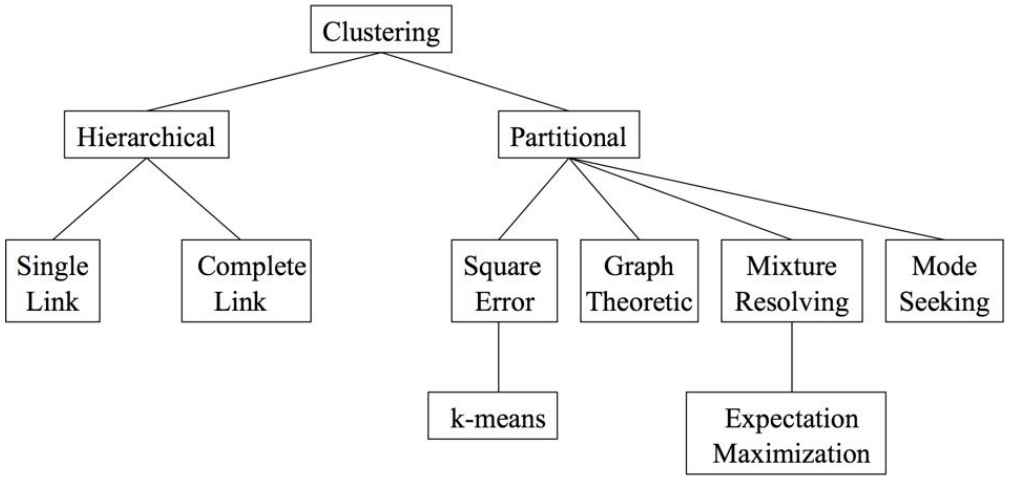

Figure 1. Taxonomy of clustering algorithms (Jain et al., 1999)

In this project we will focus on hierarchical clustering as it provides the flexibility needed to deal with both quantitative and categorical data. Hierarchical clustering transforms a dissimilarity matrix into a set of nested partitions. The result of a hierarchical clustering is usually shown as an arranged tree known as dendrogram (Murphy, 2012).

A common approach to build the hierarchical clustering is using an agglomerative algorithm. It starts assigning each element to its own cluster and then repeatedly merges pair of clusters until all elements are grouped in a single cluster. Usually, the most similar pair of clusters are

selected to be merged. There are several strategies for computing the similarity between clusters, some of the most popular approaches being single-link and complete-link.
The single-link strategy measures the similarity of two clusters by the minimum dissimilarity between pairs of items of each cluster. Differently, the complete-link strategy uses the maximum dissimilarity between pairs of items, from each cluster.

Computing the dissimilarity between instances of a data set with categorical attributes can be a challenging task. Boriah et. al (2008) review the main approaches found in the literature. A popular similarity metric for mixed data (categorical and quantitative) is the Gower coefficient (Gower, 1971), defined as the weighted mean of the contributions of each variable. Without going into detail, If x_1 and x_2 are quantitative variables, the contribution of the dissimilarity of x_1 and x_2 is given by a Minkowski dissimilarity, such as the Euclidean distance. Conversely, if x_1 and x_2 are categorical variables, the contribution of the dissimilarity of x_1 and x_2 is 1, if $x_1 \neq x_2$, or 0, otherwise.

Visualization techniques for clustered data

Heat maps have been widely used to visually represent data clustering, especially in the biological sciences. They consist of a rectangular tiling with each tile shaded on a color scale to represent the value of a cell on a data matrix. Additionally, the rows and columns of the heat map can be sorted so that similar values are close to one another. With this technique, the cluster hierarchy found on the data table is represented by a tree-like structure, called dendrogram, adjacent to the rectangle tiling (Wilkinson, 2009).

Modifications have been made to the heat map representation, in order to show different aspects of the data. Lex et al. (2010) proposed Matchmaker, a system that uses heat maps to enable the visual exploration of multidimensional data and the comparison of clustering results. The technique consists in the following pipeline: firstly, the user manually groups dimensions he is interested in analysing separately (for example, dimensions corresponding to different clustering algorithms). Then, each group is clustered separately and represented by heat maps positioned side by side. Finally, each data item is connected among the heatmaps using bundled curves. With Matchmaker, it is possible to evaluate how clustering behaves on the

dimensions of the data separately. This is mostly useful when the dimensions have a special meaning to the user.

Recently, Metsalu et al. (2015) proposed ClustVis, another heat map-based visualization for cluster analysis. In this work, the authors proposed the use of coordinated views to aid the process of data analysis: on one view, a cluster heat map is presented, which presents a lossless display of the data. On the second view, the first two principal components of the data are displayed using a scatterplot. The Principal Component Analysis preprocessing maps similar data items to dots close to each other in the scatterplot, and helps the user understand and interactively query the dataset.

Treemaps (Shneiderman, 1992) are a class of visualizations that display hierarchical structures (trees) in a space filling manner: each branch of the tree is represented by a rectangle, which is then tiled with smaller rectangles representing sub-branches. It was originally developed to solve the problem of visualizing disk usage in a computer, but has been applied to many others problems since then.

Frantz et al. (2005) used treemaps to explore large social networks. In order to explore the social network, a hierarchical clustering algorithm is used and the generated hierarchy is visualized with the traditional treemap. The visualization enabled users to easily identify important subgroups in the data, what would not be possible with traditional visualizations, such as node-link diagrams (Frantz, 2005).

Cao et al. (2011) proposed a treemap-like glyph to represent statistical properties of multidimensional clusters. Firstly, the clusters are positioned on the screen based on some attribute of the data, for example, geographical region, graph layout or similarity (computed using Multidimensional Scaling). Then, one glyph created for each cluster, which represents the composition of each cluster with a treemap layout. Two approaches are were proposed: the simpler one uses the mean value of each dimension in order to determine the size of the rectangles in the treemap. A more complex approach creates the glyphs based on the statistical distribution of the data, embedding information about kurtosis and skewness in its shape.

Some projection techniques aid in visualizing clustered data. Choo et al. (2009) proposed a two stage method for visualizing clustered data. In the first stage they applied some dimension reduction technique, such as linear discriminant analysis (LDA) (Rao, 1948) or orthogonal centroid method (OCD) (Park et al. 2009), that preserves the structure of clustered data. In the second stage they apply another dimension reduction technique, such as PCA, to transform the data to two dimensions for the purpose of visualization. They showed that using this two stage method distances within clusters are better preserved.

Self organizing maps (SOM) can be used as a dimensional reduction technique and have the advantage that they preserve the topology of the data. SOMs were used combined with some other techniques to visualize clustered data. For example, Millar et al. (2009) uses SOMs combined with Latent Dirichlet Allocation (LDA) to get a 2 dimensional plot of documents, where documents with similar topics are placed close together.

Given a 2 or 3D projection an implicit surface around items that belong to a cluster can be used to aid in the visualization of clusters. Sprenger et al. (2001) propose to construct an iso-surface using elliptic primitives on hierarchical clustered data. Balzer and Deussen (2007) used implicit surfaces based on generator points (Murakami and Ichihara, 1987) combined with transparency to show hierarchical clustered data at different levels.

Scatter plots and scatter plot matrix are very popular tools that enable the exploration of multidimensional data, and many data analysis software provide them as their basic functionality (Elmqvist, 2008). Rolling the dice (Elmqvist, 2008) is a technique that extends scatterplot matrix in the sense that it uses a dice analogy to steer the exploratory process: changing the axis of one dimension corresponds to rotating a virtual dice. In one of the case studies presented in the paper, this technique was successfully used to identify clusters in a car data set. By using the available selection tools, the user was able to see the clusters in different dimensions, corresponding to different axis configurations.

In parallel coordinates (d'Ocagne, 1885) dimensions are represented as parallel equally spaced axes. Each multidimensional item is represented as a polyline across the axes, and the position of the polyline in each axis corresponds to the value of the item for the dimension represented by such axis. However, for large data sets the visualization tends to get cluttered reducing the

possibility to analyze the data. Zhou et al. (2008) propose to use edge bundling from visual clustering in parallel coordinates. Thus, clutter is reduced and patterns in data are exposed.

Radviz, a radial mass-spring-based visualization, was also used for cluster analysis. Novakova et al. (2006) used Radviz to project multidimensional data on the screen and enable the discovery of clusters visually. The authors proposed a modification to the technique called RadizS, which projected the data to the 3D space and avoided clusters overlapping by changing the position of the points on the Z axis based on their distance to the origin.

5. Initial Mockup

We propose the following visualization for the interactive exploration of clustered data:

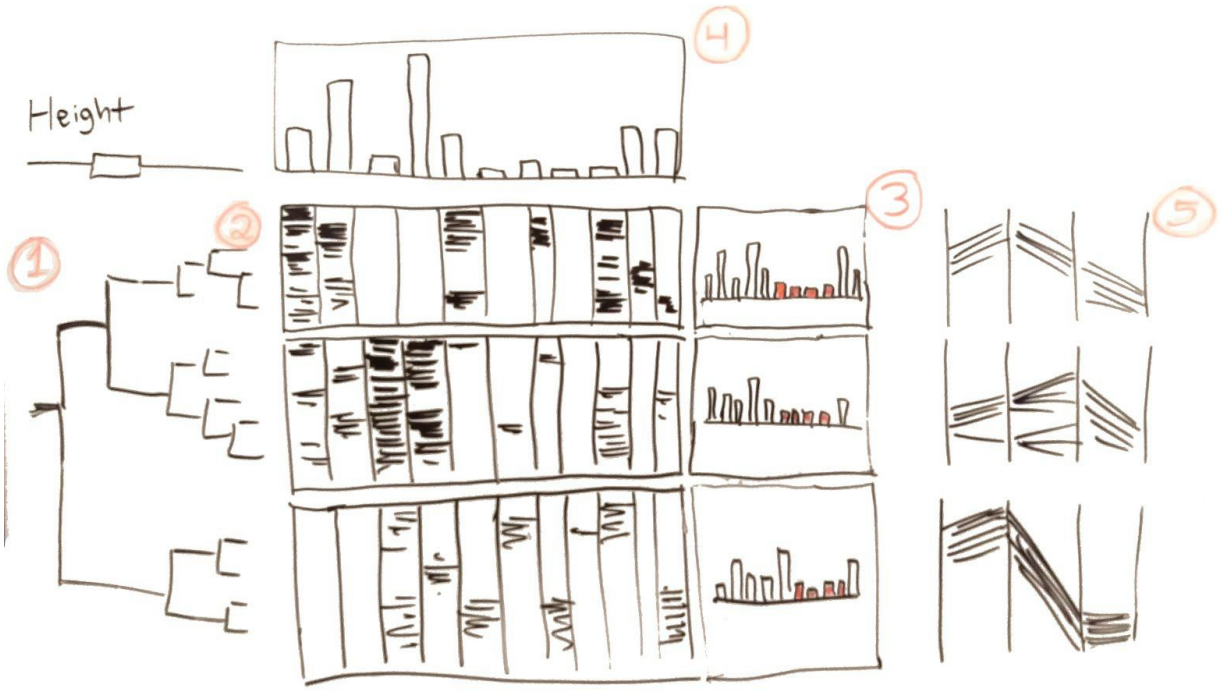

Figure 2: Initial mockup of the visualization

The visualization consists of five views, which display different aspects of the dataset:

1) Dendrogram representing the hierarchical clustering
2) Heat map representing the original data
3) Bar chart representing Relative entropy of each cluster
4) Average relative entropy of each dimension
5) Parallel coordinates of the selected dimensions

Each of the views will be described now:

1) Dendrogram representing the hierarchical clustering

We will use an hierarchical clustering algorithm to cluster the data. The dendrogram is a direct mapping of the resulting hierarchy. The user can select a cutting height to partition the data and steer the following explorations.

2) Heat map representing the original data

A heat map of the original data will be plotted beside the dendrogram. Categorical, binary and quantitative dimensions will have appropriate color schemes.

3) Relative entropy of each cluster

Shannon Entropy is a well known measure of uncertainty of data. The Shannon Entropy of a discrete probability distribution is defined by:

$$E = -\sum_{i=0}^{n} p_i \log(p_i)$$

,where *n* is the number of bins in the distribution and p_i is the probability of an element to belong to bin *i*.

Let *H* be the normalized histogram of a dimension *d*. We define the relative entropy of dimension *d* to be the entropy of *H* within the cluster (considering only the elements within the cluster), *Ed cluster*, divided by the entropy of that dimension, *Ed* (considering all the elements of the data set).

$$RelativeEntropy = \frac{E_{dcluster}}{E_d}$$

We compute the Relative Entropies of all dimensions and represent them using a bar chart. The user is encouraged to explore the dimensions with low Relative Entropy, since they are the dimensions that differentiate the most one cluster from another.

4) Average relative entropy of each dimension

We define the average relative entropy of a dimension d to be the average of the relative entropies that dimension for all clusters. On the top of the heat map (2), we represent the average relative entropies of all dimensions, aligned with the dimensions of the heat map. This enables the user to see which dimensions contributed the most to the clusterization.

The user can then sort and highlight the heat map, by moving and selecting the bins on the bar chart.

5) Parallel coordinates of the selected dimensions

For each cluster, the user can select a group of dimensions using the relative entropy bar chart (3). A parallel coordinate plot of the selected dimensions is shown, enabling the user to see a detailed view of the cluster and how similar elements of the same cluster are.

All samples will be displayed in this view, but samples corresponding to the cluster will be highlighted.

6. Project Update

After discussing with Professor Bertini, we realized that the proposed visualization did not scale well with the number of instances in the data set. For this reason, we changed our design, making a summarization of each cluster and representing it visually. Figure 3 illustrates the changes we made to the design:

Figure 3: Updated mockup of the visualization

The two changes are:

- View 5 (parallel coordinates) was removed from the visualization;
- View 2 (heat map) was replaced by a summarization of the data within the cluster. If a dimension is categorical, a stacked bar chart will encode the proportion between the categories within cluster. If the dimension is quantitative, a box plot will encode the statistical summary of the data.

With these changes, our visualization will be more scalable and will enable users to easily see an overview of the data set and the clusterization performed.

The data was clustered with the language R and package cluster. We are currently making tests with the iris and the medical data sets. Figure 4 shows the parts of the project that have been coded so far with the Iris data set:

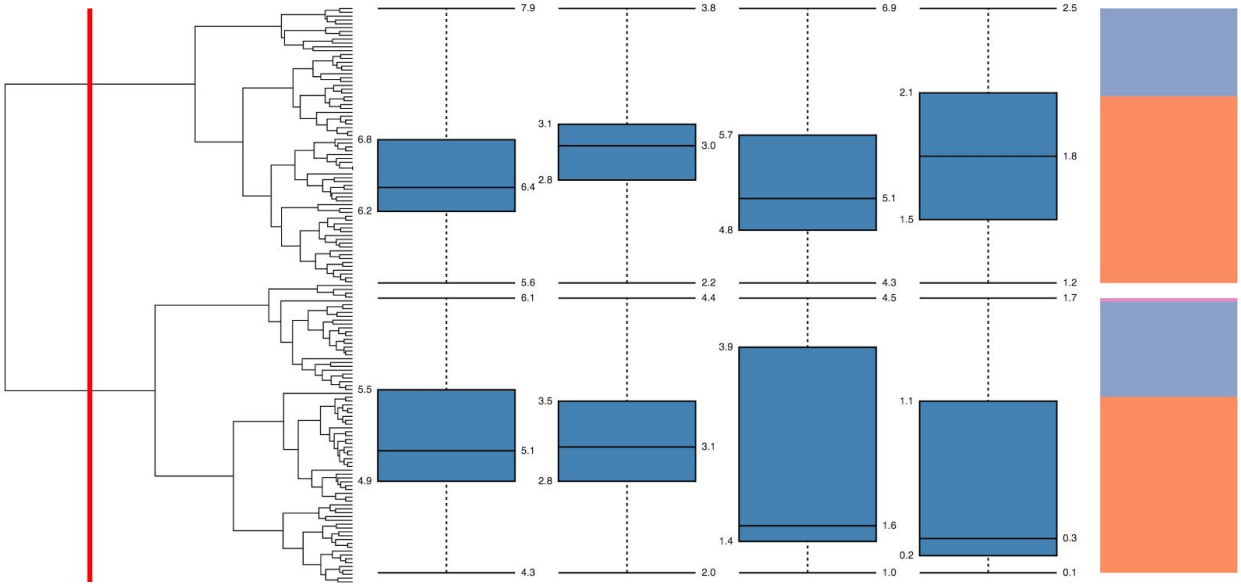

Figure 4: Features implemented in the visualization

We have implemented the dendrogram view and the tree cutting algorithm.

The user can divide the red line into smaller segments in order to clusterize the data set in different levels of details. Figure 5 illustrates this interaction mechanism:

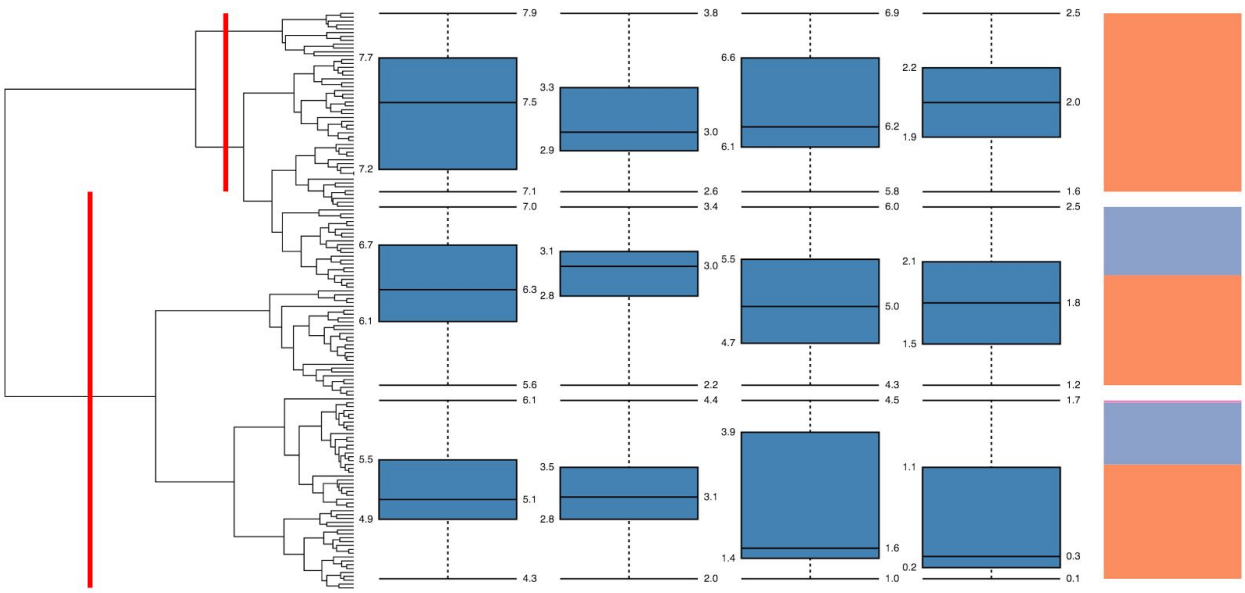

Figure 5: Interaction with the dendrogram: multiple cutting heights.

The data summarization view is currently being developed. We have implemented the stacked bar charts and the box plots. However, they are not displayed in the correct cluster position (aligned with the dendrogram). This is the next step in our implementation.

7. Final Visualization

Figure 6 shows the final visualization with the Iris data set.

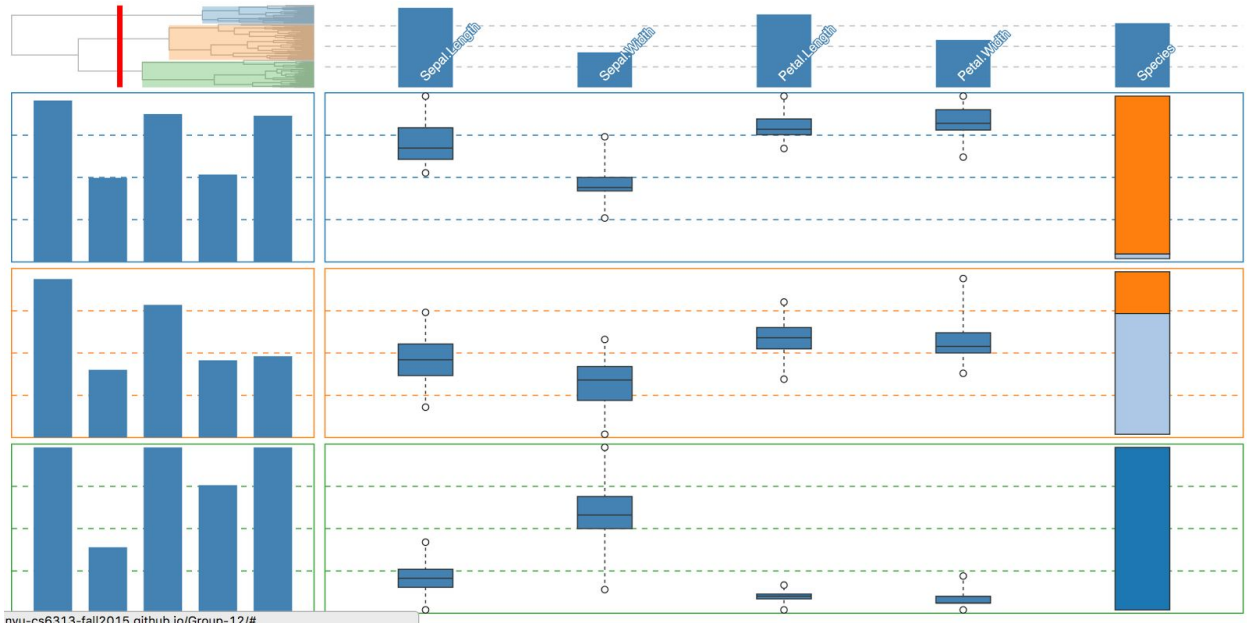

Figure 6: Final visualization with the Iris data set

In order to improve the use of space in the visualization, we have changed its layout. The dendrogram now appears in the top left corner and the relative entropy per cluster, on the left. The dendrogram now shows the number of items per cluster with an rectangular area. As in the previous update, the Data Summary view is on the center and the average relative entropy on the top.

We also changed the way the entropy per cluster is calculated. On the previous update, we used the ratio between the entropy per cluster and the total entropy of the data-set. This ratio was unfair, however, because it prioritized dimensions with fewer bins.

In order to make a fair comparison, we normalized each entropy, dividing it by the logarithm of the number of bins in base 2, and setting relative entropy to be the difference between

normalized total entropy and normalized cluster entropy. More specifically, let *n* be the number of bins in a dimension. The normalized entropy is defined as:

$$NormalizedEntropy = \frac{\sum_{i=1}^{n} p_i \log_2(p_i)}{\log_2(n)}$$

and the updated definition of Relative Entropy is:

$$RelativeEntropy = NormalizedTotalEntropy - NormalizedClusterEntropy$$

The visualization can be interpreted like in the previous updates: the user should firstly see the most important dimensions based on the bar charts, check how each dimension behave using the data summary view and update the clustering height using the dendrogram, if necessary.

8. Data Analysis

As stated in the Section 2, the goal of this project is to answer the following questions related to the medical data set:

1) Why are a set of patients grouped together by the algorithm?
2) How many types of patients are there in the population?
3) Should the government / health care provider / insurance company create new policies for the discovered patient groups?

We analysed the medical data with the proposed visualization tool and discovered some interesting facts about the clusters in the data set. Questions 1 and 2 were answered by our visualization and we believe an health specialist would be able to answer question 3 with the aid of our tool.

We now discuss some interesting findings:

1) Age and race do not influence much on the data. This can be noticed in the average relative entropy bar (Figure 7). Dimensions related to age and race have very low relative entropies. Exploring the data, we discovered that most of the patients are white, ages from 60 to 90.

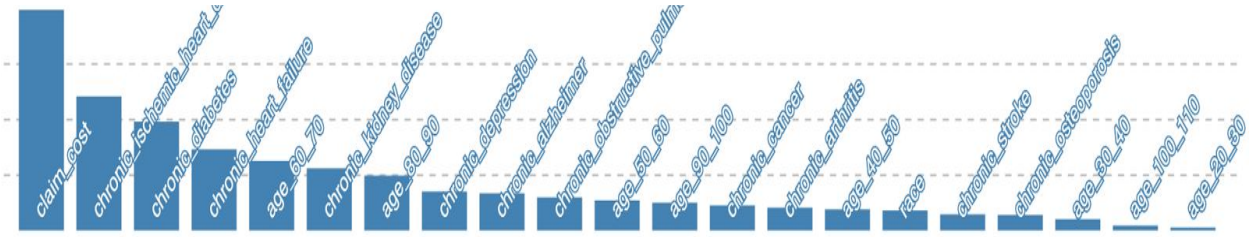

Figure 7: Average relative entropy bar chart of the medical data set

2) Dividing the records into two clusters, we obtain the visualization in Figure 8:

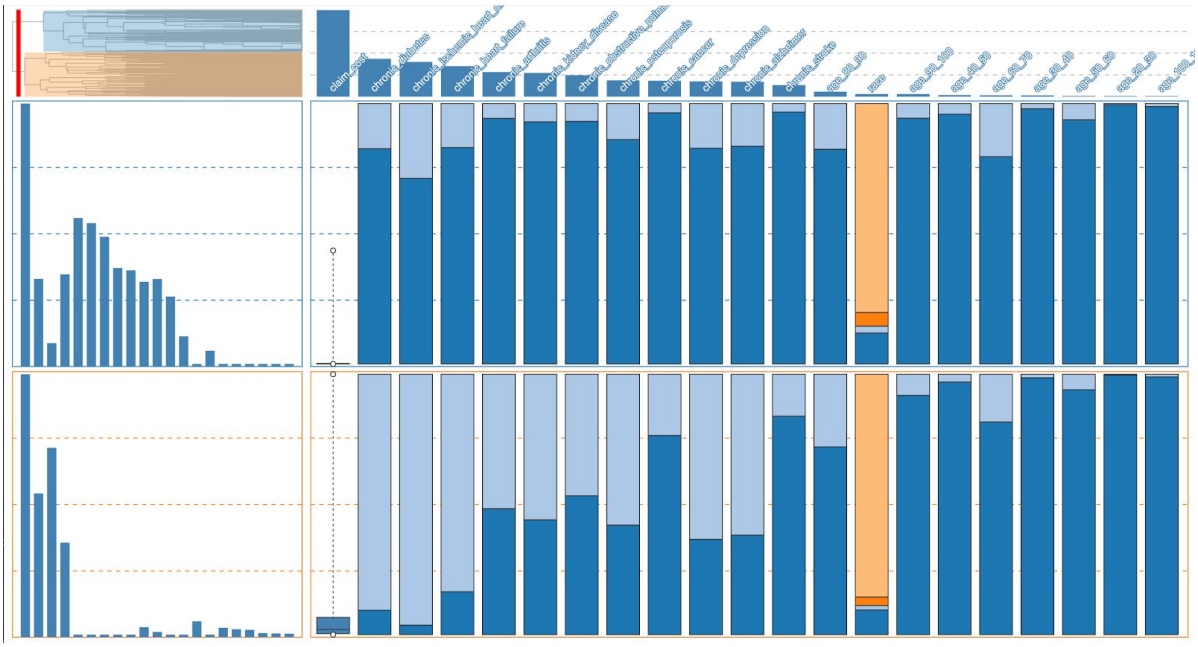

Figure 8: Division of the patients into two clusters

In the Data Summary view, dark blue color means “no” and light blue, “yes”. We notice that the patients were clustered according to their health: the set of healthy patients are in the top cluster and the set of sick patients, in the bottom cluster.

We also notice that there is a big correlation between patient health and claim cost. As expected, ill patients are a lot more expensive than healthy ones.

3) Dividing the records into three clusters, we obtain the visualization in Figure 9. We notice that the patients are now grouped into three classes: “healthy”, “ill and expensive” (median U$6000) and “ill and not expensive” (median U$2000).

A quick analysis of the data summary view indicates that the main difference between the expensive and the non-expensive patients is the presence of “chronic kidney disease”, “chronic obstructive pulmonary disease” and “chronic arthritis”. An specialist would be able to see this

pattern, form an hypothesis (for example, these are the most expensive diseases) and verify it using the original data and his own knowledge.

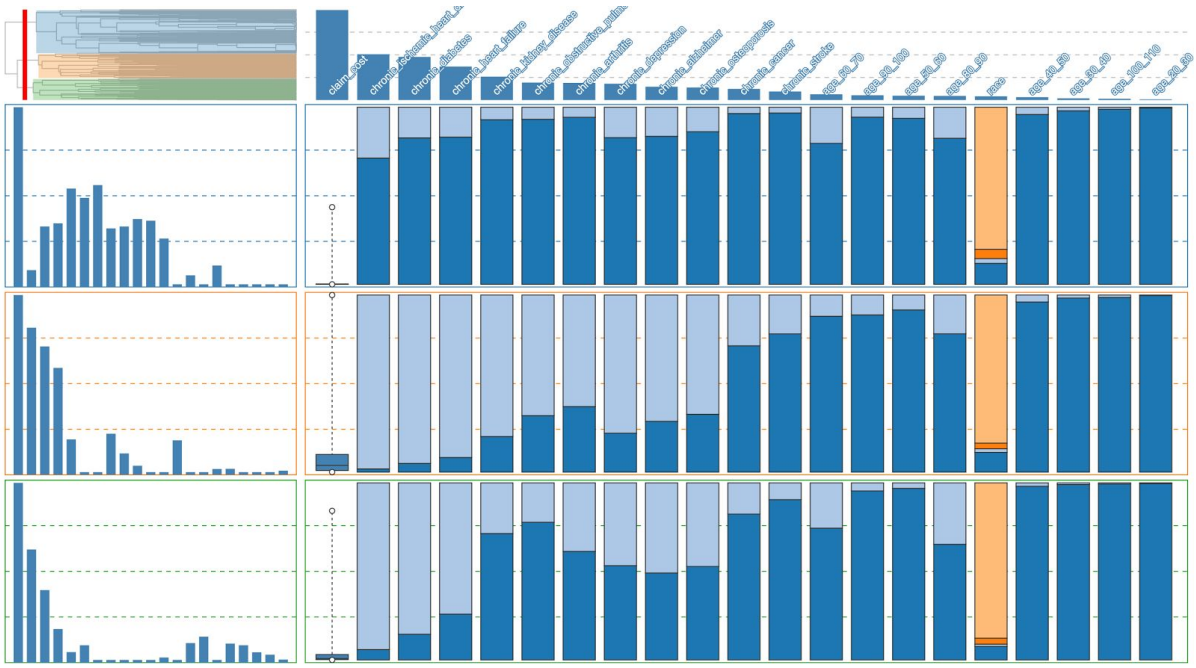

Figure 9: Division of the patients into three clusters

4) By Analysing the dendrogram, one is able to find a suitable number of clusters for the data. The number of clusters should be large enough to divide the patients into meaningful sets, but not so large that makes the analysis cumbersome. Figure 10 shows the data-set divided into 5 clusters.

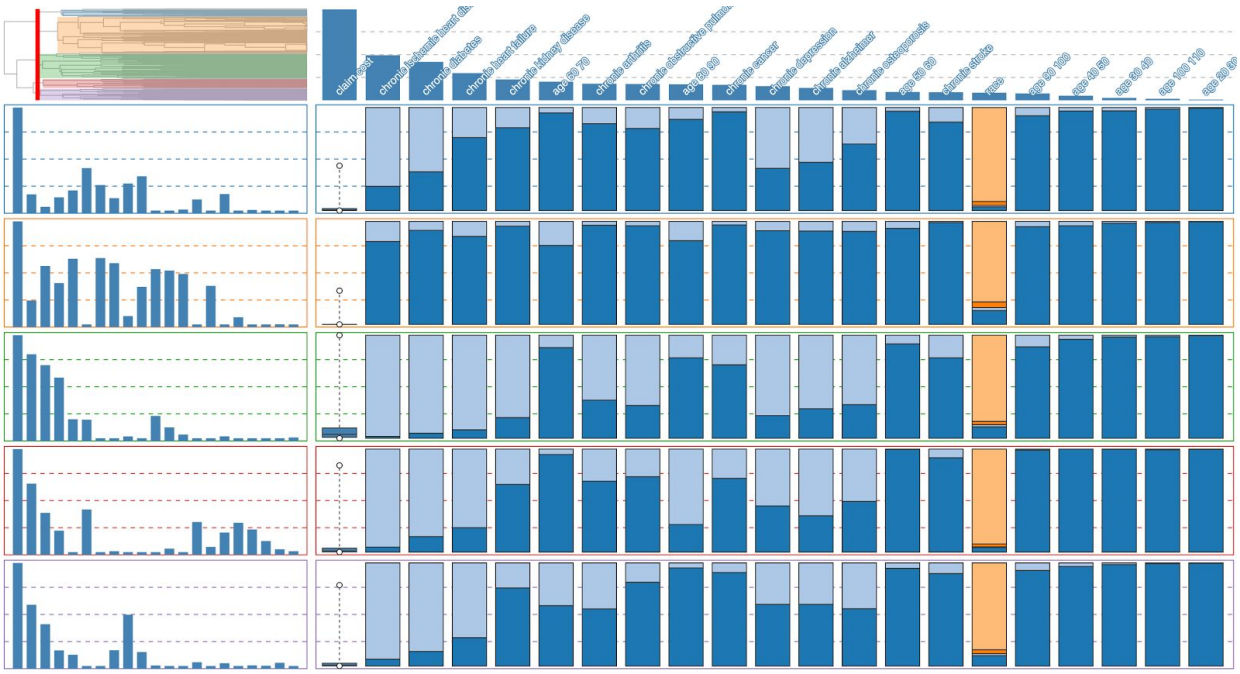

Figure 10: Division of the data set into five clusters

We now discuss our findings for each cluster:

Cluster 1 (blue) contains sick patients which are not too expensive (median: U$750). The most common diseases in this cluster are: ischemic heart disease, diabetes, depression and Alzheimer.

Cluster 2 (orange) contains healthy patients. The median claim cost is U$0.

Cluster 3 (green) contains the most expensive set of patients (median cost is U$7000). By looking at the stacked bar charts, the user can see that “chronic kidney disease”, “chronic obstructive pulmonary disease” and “chronic arthritis” are the most expensive diagnosis.

To conclude, clusters 4 (red) and 5 (purple) contains the set of patients that are sick, but are not very expensive (medians U$1900 and U$1200). The feature that distinguishes these two clusters is age: cluster 4 has older (range 80-90), and cluster 5, younger patients.

9. Limitations and Future Works

Our technique enables the rapid analysis and overview of clustered data sets. It highlights the most important dimensions for each cluster and summarizes the data using well known graphics (bar charts and box plots). Regarding the application to the medical data set, we were able to better understand how patients were clustered and why.

Our prototype seems very useful, but it has a noticeable drawback: correlations between variables cannot be represented using our current encoding, which means that this information will be lost in the visualization. As future work, we intend to investigate other approaches that enable the display of correlation, as means to improve the understanding the data.

Furthermore, we are not entirely sure that entropy is the best descriptor of importance for a dimension in the context of data clustering. We will investigate other metrics, i.e., distribution shape correlation, to try to improve the identification of relevant dimensions.

Bibliography

Balzer, Michael, and Oliver Deussen. "Level-of-detail visualization of clustered graph layouts." *Visualization, 2007. APVIS'07. 2007 6th International Asia-Pacific Symposium on*. IEEE, 2007.

Boriah, Shyam, Varun Chandola, and Vipin Kumar. "Similarity measures for categorical data: A comparative evaluation." red 30.2 (2008): 3.

Cao, Nan, et al. "Dicon: Interactive visual analysis of multidimensional clusters."Visualization and Computer Graphics, IEEE Transactions on 17.12 (2011): 2581-2590.

Cao, Nan, et al. Dicon: Interactive visual analysis of multidimensional clusters. Visualization and Computer Graphics, IEEE Transactions on 17.12 (2011): 2581-2590.

Choo, Jaegul, Shawn Bohn, and Haesun Park. "Two-stage framework for visualization of clustered high dimensional data." Visual Analytics Science and Technology, 2009. VAST 2009. IEEE Symposium on. IEEE, 2009.

Elmqvist, Niklas, Pierre Dragicevic, and Jean-Daniel Fekete. "Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation."Visualization and Computer Graphics, IEEE Transactions on 14.6 (2008): 1539-1148.

Frantz, Terrill L., and Kathleen M. Carley. Treemaps as a Tool for Social Network Analysis. No. CMU-ISRI-05-118. CARNEGIE-MELLON UNIV PITTSBURGH PA SCHOOL OF COMPUTER SCIENCE, 2005.

Gower, John C. "A general coefficient of similarity and some of its properties."Biometrics (1971): 857-871.

H. Zhou, X. Yuan, H. Qu, W. Cui, and B. Chen. Visual clustering in parallel coordinates. Computer Graphics Forum, 27(3):1047–1054, 2008.

Jain, Anil K., M. Narasimha Murty, and Patrick J. Flynn. "Data clustering: a review." ACM computing surveys (CSUR) 31.3 (1999): 264-323.

Kader, Gary D., and Mike Perry. "Variability for categorical variables." Journal of Statistics Education 15.2 (2007): 1-17.

Koren, Yehuda, and David Harel. "A two-way visualization method for clustered data." *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2003.

L. Novakova and O. Stepankova. Multidimensional clusters in RadViz. In Proceedings of WSEAS International Conference on Simulation, Modelling and Optimization, pages 470–475, 2006

Lex, Alexander, et al. "Comparative analysis of multidimensional, quantitative data." Visualization and Computer Graphics, IEEE Transactions on 16.6 (2010): 1027-1035.

M. Novotny. Visually effective information visualization of large data. In Proceedings of the Central European Seminar on Computer Graphics, pages 41–48, 2004.

Metsalu, Tauno, and Jaak Vilo. "ClustVis: a web tool for visualizing clustering of multivariate data using Principal Component Analysis and heatmap." Nucleic acids research (2015).

Millar, Jeremy R., Gilbert L. Peterson, and Michael J. Mendenhall. "Document Clustering and Visualization with Latent Dirichlet Allocation and Self-Organizing Maps." FLAIRS Conference. Vol. 21. 2009.

Murakami, S., and H. Ichihara. "On a 3D display method by metaball technique." *Transactions of the Institute of Electronics, Information and Communication Engineers J70-D* 8 (1987): 1607-1615.

Murphy, Kevin P. Machine learning: a probabilistic perspective. MIT press, 2012.

Park, Haesun, Moongu Jeon, and J. Ben Rosen. "Lower dimensional representation of text data based on centroids and least squares." BIT Numerical mathematics 43.2 (2003): 427-448.

Rao, C. Radhakrishna. "The utilization of multiple measurements in problems of biological classification." Journal of the Royal Statistical Society. Series B (Methodological) 10.2 (1948): 159-203.

Sprenger, Thomas Carl, R. Brunella, and Markus H. Gross. "H-BLOB: a hierarchical visual clustering method using implicit surfaces." *Proceedings of the conference on Visualization'00*. IEEE Computer Society Press, 2000.

Shneiderman, Ben. "Tree visualization with tree-maps: 2-d space-filling approach." ACM Transactions on graphics (TOG) 11.1 (1992): 92-99.

Van Long, Tran. "Laplacian star coordinates for visualizing multidimensional data." Computing and Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), 2013 IEEE RIVF International Conference on. IEEE, 2013.

Wilkinson, Leland, and Michael Friendly. "The history of the cluster heat map."The American Statistician 63.2 (2009).

Y. Fua, M. Ward, and E. Rundensteiner. Hierarchical parallel coordinates for exploration of large datasets. In Proceedings of IEEE Conference on Visualization, pages 43–508, 1999.

d'Ocagne, Maurice. "Coordonnees paralleles et axiales: methode de transformation geometrique et procede nouveau de calcul graphique: deduits de la consideration des coordennees paralleles/par Maurice d'Ocagne." (1885).