23rd International Conference on Knowledge-Based and Intelligent Information & Engineering Systems

Clustering and visualization of a high-dimensional diabetes dataset

Piotr Lasek\textsuperscript{a*}, Zhen Mei\textsuperscript{b}

\textsuperscript{a}University of Rzeszow, Poland, \textsuperscript{b}Manifold Data Mining Inc., Canada

Abstract

Data clustering algorithms have proved to be important and widely used methods of artificial intelligence and data mining for discovering unknown yet important patterns in datasets. Nevertheless, one of the additional aspects of data clustering is proper interpretation of the clustering results. In this paper we aim to investigate possibilities of using both data clustering and visualization methods to analyze a sample diabetes dataset. In the first part, we focus on how to cluster a highly-dimensional sample dataset and then, we concentrate on how to properly visually present the clustering results in the most meaningful way to uncover potentially interesting behavioral patterns or features of diabetes patients. In this work we examine two clustering algorithms (DBSCAN, k-Means) along with several different distance measures. We also present sample visualizations of clustering results generated by an algorithm which we have developed and discuss if the proposed way of clustering results visualization can be helpful in understanding the analyzed dataset and lead a viewer to drawing valuable conclusions about it.

© 2019 The Authors. Published by Elsevier B.V.
This is an open access article under the CC BY-NC-ND license (https://creativecommons.org/licenses/by-nc-nd/4.0/)
Peer-review under responsibility of KES International.

Keywords: data mining, data visualization, interactive data visualization, diabetes, visualization, clustering, high-dimensional clustering

1. Introduction

Chronic diseases have undoubtedly become an important aspect of today’s societies [4, 7, 8, 14, 15, 16, 19] and there is still a lot to be done to improve the ways how patients are treated and even more to better understand how the

\textsuperscript{*} Corresponding author. Tel.: +4-817-851-8531; fax: +4-817-851-8525.
\textit{E-mail address:} lasek@ur.edu.pl
diseases spread and what the most important decisive factors in those diseases are. For example, in Canada, there are eleven million people with one or more chronic conditions (heart diseases, diabetes, asthma, arthritis, cancer, depression, etc). Those patients are not only responsible for the great majority of the total health care cost but also require more physician visits, hospital admissions, total hospital days, prescriptions, and home care visits than an average patient.

Obviously, a number of methods for analyzing and discovering unknown yet interesting patterns in medical datasets is known and new ones are continuously proposed [20, 21]. Nevertheless, complexity and larger sizes of datasets require developing more interactive and easy-to-understand ways of presenting the results of algorithms to analysts. Interactive exploration and visualization become a must in the today’s world and the statement that one picture is worth a thousand words is nowadays even more up to date. Interactive data visualization is an important bridge connecting data with people. It can not only dramatically (if done correctly) improve the possibilities of understanding data, it can actually turn data into knowledge. It has been even suggested recently [1] that scientists might be better off analyzing images than of actual data.

Data visualization has a long history behind [22] and can be applied to any step of data analysis. Nevertheless, for the purpose of this work we will focus on applying visualization techniques for meaningful presentation of the results of clustering high-dimensional datasets. The main goal of this work is to determine if by means of clustering and proper visualization it is possible to discover behavioral patterns or features of diabetes patients. So, we aim at finding ways of presenting the results of clustering so it can let viewers to easily understand the clustering results and draw valuable conclusions about the dataset.

Organization of the paper is as follows. In Section 2 we remind two of the most popular clustering algorithms: a density-based clustering algorithm DBSCAN [3] and a well-known k-Means [9]. In Section 3 we describe several clustering and visualization experiments which led us to implementation of an application for visualization of high-dimensional (with over 1200 attributes) dataset. We summarize the results, conclude the paper and discuss further steps in the final section.

2. Data clustering and visualization

2.1. Data clustering

Clustering data into meaningful groups has always been an important task of both artificial intelligence and data mining. Clustering is considered as an unsupervised classification of data where the results of the task depend on the algorithm used. A number of different clustering algorithms have been offered in the literature over time. Some of them are capable of discovering proper clustering of data only when the number of clusters is known in advance. Other algorithms can discover clusters of particular shapes only. There are also algorithms that are able to identify noise data. In this subsection we remind two commonly used algorithms, namely: DBSCAN and k-Means.

DBSCAN. The main feature of this algorithm is that each point of a cluster must contain at least a certain number of points (MinPts) within its \( \epsilon \)-neighborhood (\( \epsilon \)). In other words, the density in the \( \epsilon \)-neighborhood of a point belonging to a cluster must be greater or equal to a predefined threshold. The clustering process in DBSCAN is based on the following concepts of relations between points: directly density-reachability and density-reachability (please refer to [3] for the definitions and detailed descriptions of those concepts). DBSCAN discerns three types of points: core points, border points and noise points. A cluster in the context of the DBSCAN algorithm is a region of high density. Regions of low density constitute noise. A point in space is considered a member of a cluster if there is a sufficient number of points within a given distance from it. Firstly, the algorithm generates a label for the first cluster to be found. Next, the points from the dataset are processed. The initial value of the ClusterId attribute of the first point read is equal to UNCLASSIFIED. While the algorithm analyzes point after point, it may occur that the ClusterId attributes of some points may change before these points are actually analyzed. Such a case may occur when a point is density-reachable from a core point examined earlier. Such density-reachable points will be assigned to the cluster of a core point and will not be analyzed later. If a currently analyzed point has not been classified yet (the value of its ClusterId attribute is equal to UNCLASSIFIED), then the ExpandCluster function is called for this point. If the point...
is a core point, then all points within its \(\varepsilon\)-neighborhood are assigned by the \textit{ExpandCluster} function to the cluster with a label equal to the current cluster’s label. Next, a new cluster label is generated. Otherwise, if the point is not a core point, the attribute \textit{ClusterId} of point \(p\) is set to \text{NOISE}, which means that the point will be tentatively treated as noise. After analyzing all points in the dataset, each point’s attribute \textit{ClusterId} will store a respective cluster’s label or its value will be equal to \text{NOISE}.

\textbf{K-Means.} The k-Means algorithm is the simplest and most known representative of a group of minimum-variance or partition algorithms for which the goal of the algorithm is to minimize the sum of squared error criterion function \cite{11} and the number of clusters is known. Partitioning clustering algorithms produce single data partitions instead of creating a structure such as dendrogram created by hierarchical clustering algorithms. A major problem related to the fact that a user must pre-set a number of clusters to determine with partitioning algorithms is actually a proper selection an appropriate number of output clusters \cite{5}. In k-Means algorithm, each cluster is represented by the gravity center of the cluster, so called centroid. Analogously, in a similar partitioning algorithm, k-Medoids \cite{6}, each cluster is represented by its center point belonging to a dataset called medoid. Another example of the partitioning clustering algorithms is an improved version of k-Medoids - CLARANS \cite{2}.

To process a dataset, the k-Means algorithm starts with a first group of \(k\) randomly selected centroids. Those centroids are used as initial center points for every cluster. Then, the algorithm, iteratively performs calculations in order to optimize the locations of centroids. The calculations are stopped when either the centroids locations have stabilized (this means that their coordinates do not change between iterations anymore; in other words, the clustering has been successful) or the pre-defined number of iterations has been achieved.

\textbf{2.2. Data visualization}

Interactive data visualization, as the Gartner’s report confirms \cite{13} is an important need of today’s data exploration and discovery tools. The process of data visualization is often described in the literature by the Schneiderman’s famous mantra: \textit{Overview first, zoom and filter, details on demand} \cite{26}. This means that the analyst is primarily concerned with obtaining a general visual description of the analyzed data set. In the next step, he looks for specific and interesting details. While in the case of small data sets, visualization of them is practically not a problem, then in the case of large datasets or data with a large number of dimensions you can no longer assume that you only need to display the original set of data on the screen - the number of objects can be just so large that it will exceed the number of available pixels by orders of magnitude \cite{10}. In other words, the data has become too large to be able to see it directly. However, data visualization clearly becomes a final yet a critical step in extracting knowledge from data which is confirmed by recent calls to action \cite{24}.

Nevertheless, some of the data visualization challenges have already been and still are being addressed \cite{22}. For example, if we are focusing on how to efficiently visualize data we can use techniques of data reduction such as summarizing (e.g. employing data clustering), condensing (using so-called binned aggregation) or bounding (visualization by the number of pixels used to display it, e.g. nanocubes). Similarly, we can try to reduce data to be visualized before it is even sent to a visualization client. Here, filtering, aggregation and sampling come in useful.

\textbf{2.3. Data presentation challenges}

In this work we are employing three of the above-mentioned methods of dealing with large amounts of data objects to visualize, namely: clustering (for summarization of data points), aggregation (for generating descriptions of data clusters) and visual condensing (for generating more meaningful visualizations).

\textbf{Clustering.} For clustering we employ the k-Means and DBSCAN algorithms to label points with cluster ids. We do also examine several different distance measures (Euclidean, Manhattan, Tanimoto). Discovered clusters are then aggregated and condensed to generate a meaningful visual representation of a dataset.

\textbf{Aggregation.} We aggregate points assigned to clusters by calculating attribute-value descriptors representing how often specific values of attributes occur in a cluster. This is later drawn in a visual form of bars where values are represented by lines of length corresponding to how frequently they occur in a cluster.
Condensing. We plot the results of clustering and aggregation using a special form of a bar-chart where bars represent frequency of values of attributes. This approach allows us to efficiently represent a large number of attributes showing at the same time aggregated distribution of values of a given attribute within a cluster. Additionally, we can plot several clusters in the same figure so that one can easily visually compare the clusters. Such a way of visualization makes possible to quickly determine substantial or outlying attributes defining a cluster. Based on this an analyst can draw conclusions and gain knowledge about the dataset’s objects.

3. Experiments

In this section we start with a description of the dataset we have used for our experiments. Then, we describe three groups of experiments that we have conducted. First, we tried to find a way of showing the results of clustering of a dataset with over 1200 dimensions. Then, by means of clustering, we tried to confirm a hypothesis concerned with a manual way of segmenting the dataset. Finally, we present a method employing DBSCAN (with the Tanimoto distance measure) and our visualization system for presenting and analyzing high-dimensional results of clustering.

3.1. The dataset

The original dataset we use contains 65,000 of records from a 2010 Canadian Community Health survey†. The dataset includes information on various aspects of people’s life such as lifestyle factors and socio-demographics, attitudes, stress level, satisfaction, exercise, diet and smoking (Figure 1). The attributes were of different types: e.g. numerical, nominal, intervals. In some cases, we focused on analyzing a subset of the original dataset containing about 17,000 object representing different patients with diabetes and heart disease. In this case each patient was described using about 30 – 200 attributes, both numerical, nominal or ordinal. The main goal of the analysis was to try to find groups of diabetes patients and determine what the main factors and attributes characterizing them were.

3.2. Mapping all attributes to numerical values

In our initial experiment we tried to overcome the issue concerned with different types of attributes. We prepared a procedure for mapping all types of attributes to numerical values from the interval [0, 10]. For example, Boolean values like ‘Yes’ and ‘No’ were mapped to integer values such as 10 and 0 respectively. For nominal values, such as

† [Link](http://www23.statcan.gc.ca/imdb/p2SV.pl?Function=getSurvey&Id=81424)
for example ‘In person’, ‘On telephone’, ‘N/A’ we to assigned integer values which would be as adequate as possible. In this case, the mapped values were 10, 5 and 0. Further, for intervals we assigned values also based on our preliminary analysis, e.g. to an age interval ‘0 – 10 years old’ we assigned 0, to ‘11 – 20 years old’ we assigned 1, to ‘21 – 30 years old’ we assigned 2, etc.

<table>
<thead>
<tr>
<th>Attribute group</th>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADL</td>
<td>ADL_01</td>
<td>Needs help - preparing meals</td>
</tr>
<tr>
<td>ADL</td>
<td>ADL_02</td>
<td>Needs help - getting to appointments</td>
</tr>
<tr>
<td>ADL</td>
<td>ADL_03</td>
<td>Needs help - doing housework</td>
</tr>
<tr>
<td>ADL</td>
<td>ADL_04</td>
<td>Needs help - personal care</td>
</tr>
<tr>
<td>ADL</td>
<td>ADL_05</td>
<td>Needs help - moving about inside house</td>
</tr>
<tr>
<td>ADL</td>
<td>ADL_06</td>
<td>Needs help - looking after finances</td>
</tr>
<tr>
<td>ADL</td>
<td>ADLF6R</td>
<td>Help needed for tasks - (F)</td>
</tr>
<tr>
<td>ADM</td>
<td>ADM_N09</td>
<td>Interview by telephone/in person</td>
</tr>
<tr>
<td>ADM</td>
<td>ADM_N10</td>
<td>Respondent alone during interview</td>
</tr>
<tr>
<td>ADM</td>
<td>ADM_N11</td>
<td>Answers aff./presence of another person</td>
</tr>
<tr>
<td>ADM</td>
<td>ADM_PRX</td>
<td>completed by proxy</td>
</tr>
<tr>
<td>ADM</td>
<td>ADM_RNO</td>
<td>Sequential record number</td>
</tr>
</tbody>
</table>

Fig. 2. Two sample groups of attributes. First (ADL) is a group of attributes representing different types of help needed (preparing means, getting to appointments, doing housework, personal care, etc.), the latter is a group corresponding to a way how the survey was filled in (by a respondent himself or with a help of other people or systems).

When the attributes are of different types and when they are numerous, it is difficult to properly choose a distance measure for the clustering algorithm. However, mapping all types of attributes into numerical ones, allowed us to use the Tanimoto measure – a measure which finds numerous applications e.g. in bio- and chemical-informatics, but also in web and text mining. The Tanimoto similarity measure proved to be very efficient for calculating distances between high-dimensional representations of objects [25]. The measure for two vectors \( u \) and \( v \) can be calculated using the following formula:

\[
T(u, v) = \frac{u \cdot v}{u \cdot u + v \cdot v - u \cdot v}
\]

For the preliminary experiments we used the DBSCAN algorithm. We run it several times so that we could adjust values of MinPts and Eps parameters. The number of clusters changed from 8 to 13. Then, we wanted to visualize the results of clustering. However, visual analysis of more than 1200 attributes simply plotted in a form of graphical bars would not result in providing an analyst with a visualization easy to understand and interpret. Thus, we rather focused on how to simplify (condense) the visual representation of the clustering results and it turned out, that in the analyzed dataset attributes can actually be grouped into segments of similar attributes (We show two sample groups of similar attributes in Figure 2). This observation allowed us to condense data to be visualized even more and allowed us to reduce the number of graphical objects representing a single class at least by one order of magnitude. We have managed to group attributes belonging to similar categories and calculate average values of attributes within attributes groups. This led to the reduction of dimensions to be shown to c.a. 100 dimensions.

The visualized results of three sample clustering experiments can be found in Figure 3. The DBSCAN parameters we used differed from 0.3 to 0.5 (Eps) and from 15 to 25 (MinPts). As one can see, there are groups of attributes (represented by single colored bars) in each of the clusters which clearly stand out (compared to the same attribute
groups from different clusters). Moreover, one can easily point which features make the cluster distinct from the other clusters by visually comparing graphical patterns of groups of similar attributes (bars). For the number of discovered clusters which is relatively small (not much greater than 10) this is an easy task and can lead the viewer to understand which features or attributes constitute particular clusters.

Table 1. The mapping of two chosen attributes (self-perceived health and self-perceived health – compared to 1 year ago) to a new attribute – a segment of a patient (Anxious, Depressed, Gloomy, Happy, Optimistic, Rejuvenated, Satisfied).

Table 2. The comparison of clusters generated by DBSCAN and k-Means to test if the manual segmentation (by means of the additional attribute - the patient’s segment defined in Table 1) reconciles with the results of clustering. In the tables below, each cluster has been split into rows corresponding to numbers of objects in each of the manually created segments.
3.3. Finding correlation between segments and clusters

The goal of this experiment was to check whether the manual segmentation based on a new attribute (the patient’s segment) (Table 1) reconciles with the assignment of patients to clusters generated by the clustering algorithms. The intuition behind introducing this attribute was that the patients are able to describe by themselves the state of their health quite well (especially when comparing to how they felt a year ago).

In this case we have performed the experiments using only 34 chosen attributes (selected by statistical importance of those attributes). The algorithms we used were DBSCAN with the Tanimoto measure and k-Means with Euclidean distance measures. Regardless of initial intuition, the results we obtained did not confirm that such a correlation exists. In each of the discovered groups we could see representatives of all of the segments which means that the segments were not in line with any of the clustering results we have obtained. We came to the same conclusions clustering the data set using only 7 selected attributes and the same algorithms with similar settings. The results has been presented in Table 2.

3.4. High-dimensional visualization

In this series of experiments, we have decided not to preprocess the dataset but to use all dimensions available. Additionally, we did not try to map the values of nominal and interval attributes into numerical ones. Instead we have employed a special distance measure which can be used to calculate distances between objects described with attributes of different types. However, we have limited our experiments to the patients with diabetes and heart diseases as those two often occur together. Below we describe the details of the experiments starting from the distance measure we used, parameters of the clustering algorithm. Then we describe the system we have implemented and discuss its possibilities for data visualization of high dimensional datasets.

Distance measure. The distance measure (\(d_{h,i}(u, v)\)) we used to handle multiple types of attributes can be express with the following formula:

\[
d_{h,i}(u, v) = \begin{cases} 
0 & \text{if } u_i \text{ and } v_i \text{ are nominal and } u_i = v_i \\
1 & \text{if } u_i \text{ and } v_i \text{ are nominal and } u_i \neq v_i \\
|n(u_i) - n(v_i)| & \text{if } u_i \text{ and } v_i \text{ are continuous}
\end{cases}
\]

where \(m\) is the number of attributes.

Clustering with DBSCAN. We used DBSCAN to cluster dataset. This time we have performed a number of experiments with different values of MinPts and Eps parameters of DBSCAN (using similar values to those used in the experiments described earlier). However, in this case, we used more attributes, starting from 37 selected ones (based on their statistical significance) up to all 1207 attributes. Also, we did not try to reduce the number of attributes to make clustering results visualizations easier to understand. Instead, we have implemented a visualization client capable of presenting all attributes at the same time.

The visualization client. The application we have developed is capable of presenting all the attributes using for clustering in one single visualization. It may seem, and we have actually pointed this out earlier, that a visualization of a large number of attributes can be difficult to analyze but we have tried to visualize them so that the viewer would actually able to visually analyze them easily (Figure 4). First, every cluster is represented by a set of bars (attributes). Each bar represents a single attribute which is then divided into sub-bars, where each sub-bar corresponds to a single value of an attribute (the length of the sub-bar is calculated based on how often the particular value occurs within objects assigned to the cluster). Additionally, when a user moves the cursor over a particular bar (attribute) or a sub-
bar (value), the application display the name of the attribute and its value over which the cursor is located). The user of the application can easily configure a subset of attributes to display, run the clustering with different parameters, save current experiment into a database as well as open previously saved clustering results as a visualization. The application is available for cloning from a GitHub repository: https://github.com/piotrlasek/cv-hd.

![Figure 4. A sample visualization of clustering of a high-dimensional dataset.](image)

<table>
<thead>
<tr>
<th>Table 3. Visual interpretation of clustering results (experiment 3.4.1).</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster</td>
</tr>
<tr>
<td>---------</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 4. Visual interpretation of clustering results (experiment 3.4.2).</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster</td>
</tr>
<tr>
<td>---------</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 5. Visual interpretation of clustering results (experiment 3.4.3).</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster</td>
</tr>
<tr>
<td>---------</td>
</tr>
<tr>
<td>1</td>
</tr>
</tbody>
</table>
Interpretation of results. In Tables 3-4 we have collected several findings based on visual analysis of sample clustering results of three chose experiments which we have conducted. The results of the analysis seem to provide valuable knowledge about the analyzed dataset. The algorithm discovered that, depending on the values of parameters we used, there exist several different groups of diabetes patients which can be characterized with several different attributes. Some of the interesting examples are:

- There exists a group of older people in Ontario whose main source of personal income is senior benefit, whose perceived health is poor (Table 3).
- High daily energy expenditure and good or very good perceived health are characteristic features of diabetes patients from Alberta with income higher than 80000 dollars (Table 4).
- Patients who live in the Quebec province, speak French only, are married or widowed, do not drink, perceive their health as good or very good (Table 5).

4. Conclusions and further works

In this paper we have aimed at investigating possibilities of using data clustering and visualization methods to analyze a sample high-dimensional diabetes dataset. We have performed a number of clustering experiments using two commonly known clustering algorithms (DBSCAN, k-Means) with different distance measures (Euclidean, Manhattan, Nominal-Numeric). We have also experimented and proposed a method for visual analysis of clustering results using our developed visualization application which main feature is that it is capable of visualizing results of clustering of high-dimensional datasets so that the graphical representation is not disturbed with additional information (attributes names or values) and the viewer can focus on analyzing the general view of clustering results but can get detailed information on interesting attributes according to the known Schneiderman’s mantra: Overview first, zoom and filter, details on demand.

Our further works will aim at further development of the visualization application and focusing on implementing and experimenting with new methods of interactive data exploration and visualization.

References


[14] Partnering with patients to drive shared decisions, better value and care improvement. Workshop Proceedings. Institute of Medicine, August 15, 2013


