Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 112 (2017) 1747–1754

Procedia Computer Science

www.elsevier.com/locate/procedia

21th International Conference on Knowledge Based and Intelligent Information and Engineering Systems, KES2017, 6-8 September 2017, Marseille, France

Visual instance-based recommendation system for medical data mining

Joris Falip[a,*], Amine Aït-Younes[a], Frédéric Blanchard[a], Brigitte Delemer[b], Alpha Diallo[b], Michel Herbin[a]

[a] *CReSTIC, University of Reims Champagne-Ardenne, Reims, France*
[b] *Department of Endocrinology, Diabetes and Nutrition, University Hospital of Reims, Reims, France*

Abstract

This paper presents an instance-based algorithm allowing exploration of large medical dataset by making pairwise connection between patients. In our metric-free method, each individual in a dataset ranks every member of the dataset. By aggregating these ranks, it is then possible to visualize data according to typical individuals representing subsets of closely-related patients. The paper also describes a visualization tool allowing exploration of a database of diabetic patients. This prototype of a recommendation system implements the aforementioned algorithm to enrich data, structure patients, create associations between individuals and provide recommendations.

Keywords: Exploration ; Recommendation ; Medical Data ; Instance-based learning

1. Introduction

As electronic health records and wearable sensors become more widespread, medical datasets tend to be larger and call for specific methods of exploration. These datasets come with inherent problems : they contain high-dimensional data[1] which can be heterogeneous and unstructured, often including missing values. With such amount of available data, doctors need assistance to find relevant patients. They rely on recommendation systems to browse[2,3], explore and manipulate records, allowing them to find links between similar patients. Despite machine learning applications to medical data showing promising results[4], use of traditional approaches does not exactly fit the clinical context[5]. These approaches tend to generalize and exclude outliers, algorithms need to be trained and most solutions prove to be "black boxes" that reduce interpretability[6,7]. While they perform well when trying to predict diseases, those algorithms are not suitable when trying to understand the same diseases and study atypical patients without heavily relying on domain expertise. The purpose of this paper is to provide a method able to structure elements of a dataset by creating associations between them. The resulting structure enriches data with measures of representativeness[8],

[*] Corresponding author. Tel.: +33 3 26 91 84 58.
E-mail address: joris.falip@univ-reims.fr

1877-0509 © 2017 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of KES International
10.1016/j.procs.2017.08.205

and is a way to visualize typical individuals or similar patients in a medical database. The proposed algorithm is instance-based, as to avoid overgeneralization. It is also a mean of simulating the reasoning of doctors faced with new patients : they match new individuals with past cases they encountered. Another advantage of this method is the ease with which a user can express constraints on the association rules between elements, making it a customizable tool. The main component of our method is an election algorithm where individuals attribute a score to every member of the dataset, in order to determine the most representative individuals across the population (this concept is close to class prototypes, but does not need any prior clustering step on the dataset[9]). By using different functions to calculate the score of each individual, the output structure can easily be modified.

This contribution is divided into the following sections : first we share some background related to our work, the problems we tackle and the data we use to illustrate this article. The second section focuses on concepts and algorithms that enrich data and create recommendations. The following section describes in details the visualization and exploration tool we built using the aforementioned algorithms. This paper ends with a discussion and presentation of future related works.

2. Problem and dataset

As we mentioned previously, deep learning and traditional approaches does not fully meet our criteria. Firstly, predicting the occurence of a disease does not necessarily help to understand its origin. By matching the studied patient with other closely-related and similar individuals, it becomes easier for a doctor to picture the evolution of the patient's condition. In addition, our aim is to stay as close as possible to the way doctors analyze medical records : they rely on their experience to associate similar patients, and use their knowledge of those patients to treat a new individual. The objective of the proposed method is to help doctors to enhance their knowledge by automatically suggesting similar patients and allowing to visualize all the gathered data on those same patients. Our goal is to use the way medical staff analyze patients records and apply it to the large clinical datasets hospitals gather. To do this, the tools we develop should favor emergence of new hypothesis and extraction of original and relevant associations and patterns.

The dataset we are working on is extracted from shared electronic medical records of diabetic patients. These anonymized data come from the database of a regional health network : *CARéDIAB*. Using these records calls for a lot of preprocessing, such as merging redundant records, inferring missing values and fixing incorrect values. Moreover, the study focusing on complications related to diabetes, the dataset needs to be filtered and curated. Once this is done, the resulting dataset contains information on 28073 medical procedures associated to 93 medical tests, involving 1437 patients.

3. Exploring data through visualization and recommendation

Based on social choice theory, the proposed algorithm combine individual preferences in a "representativeness score" given to each entity. Patients will then explore their neighborhood and choose the neighbor with the highest score as their representative. Multiple individuals with the same representative will therefore be closely-related patients. Concept of representativeness, as used in this work, was established in a previous paper. In order to stay close to how doctors reason, the method presented in this section does not make any assumption regarding a number of classes in which patients are divided, and does not penalize nor exclude patients based on their euclidian distance in relation to the rest of the dataset.

3.1. Workflow

In order to assist doctors in diagnosing their patients, we aim to deliver a full framework for a recommendation system oriented toward clinical data. The workflow of this tool is described in Fig. 1. This paper focuses on the structuring step, and the algorithm is detailed in the next subsection, along with examples using synthetic and real datasets.

The filtering phase of the process was covered in a previous work. Most of the filtering consists in homogenizing input data into a structure akin to a relational database. To achieve this, multiple snapshots of a patient health status

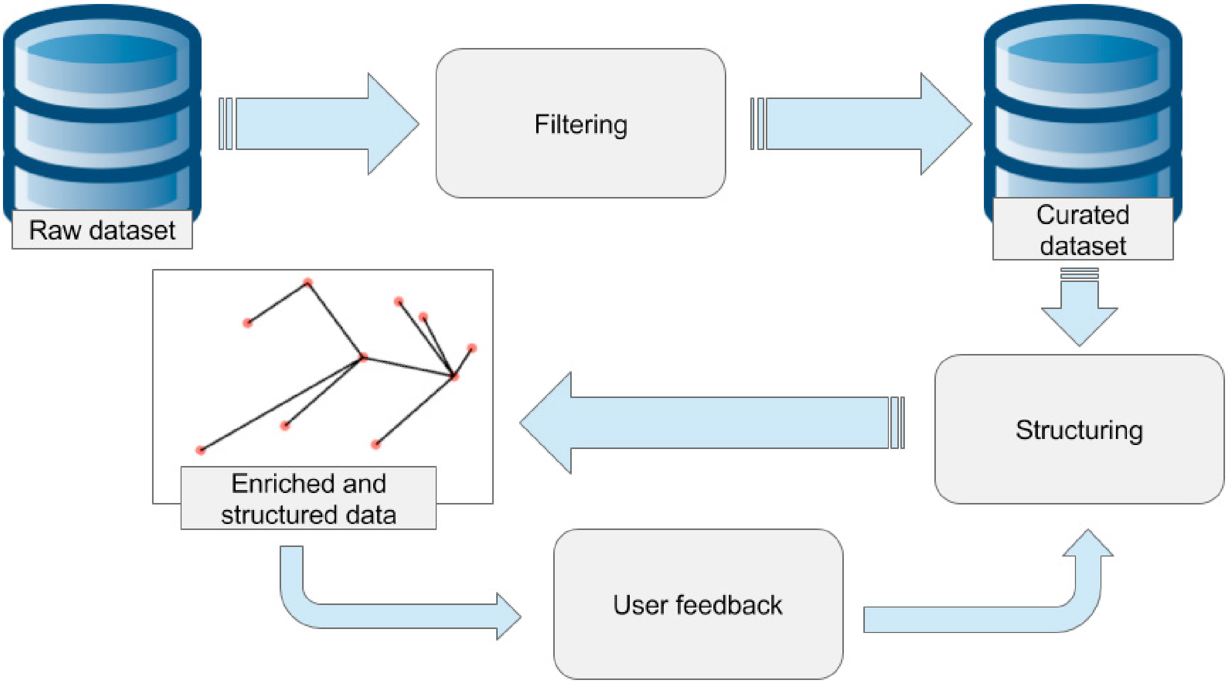


Fig. 1. Workflow of the various steps included in our prototype of a recommendation system.

across several years are gathered and merged together in time series. This step, of course, will greatly vary depending on how input data are structured, but its goal is to store knowledge in a practical form.

Taking the user feedback into account, while not yet implemented in our prototype, is a key element to the process. If users can rate an association between patients as "meaningful" or "incorrect", this can be used to guide the structuring process in order to deliver better recommendations. Feedback can influence structuring in multiple ways, such as weighting each dimension to give more importance to specific criteria, or using a veto to specify particular dimensions that should not be taken into account when matching individuals. Implementing user feedback regarding the proposed recommendations allows users to train the system and guide it toward improved and more accurate results.

3.2. Algorithm

The whole dataset is structured according to the *Degree of Representativeness* (*DoR*). The DoR acts as a measure of importance for each object in the set, and determines to what extent this object can be an exemplar. It is computed by aggregating scores given by every member of the dataset, on each dimension. The algorithm detailed below is illustrated by Fig. 2

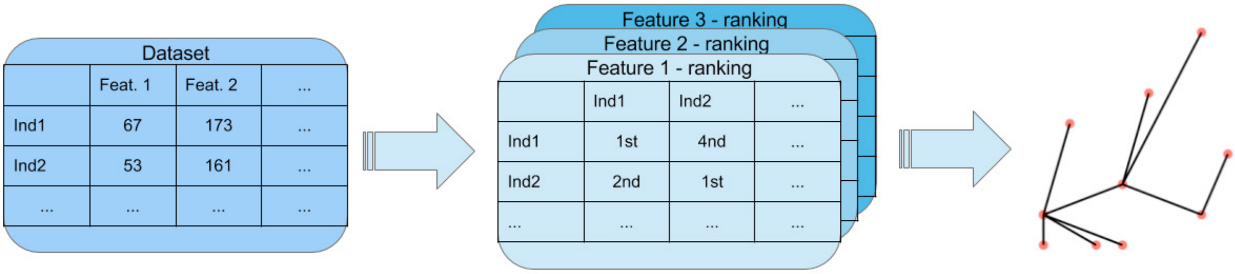
Let Ω be a set of n elements in a multidimensional space. This n elements are called objects, and consist of D qualitative or quantitative features.

Let us define the ranks needed to compute the DoR of each object. For each of the D dimensions, we compute a dissimilarity matrix for the given dimension. Each object then ranks every object according to their similarity, resulting in a rank matrix. This is done for each of the D dimensions.

Let us transform these ranks into scores. Let x be an object of Ω : x assigns a relative score Sc_{xD} to all objects of Ω for dimension D. The score Sc_{xD} relative to x can be any arbitrary function, but in this paper it will be defined by :

$$\forall y \in \Omega, Sc_{xD}(y) = n - Rk_{xD}(y) \tag{1}$$

where $Rk_{xD}(y)$ is the rank of an object y relative to x on dimension D. Computing all relative scores, each object x receives $n * D$ scores, corresponding to the votes of all n objects of Ω on all D dimensions. All these $n * D$ relative scores are aggregated to define the *DoR* of x. The aggregate score is defined by:



Dataset

	Feat. 1	Feat. 2	...
Ind1	67	173	...
Ind2	53	161	...
...	...	...	...

	Ind1	Ind2	...
Ind1	1st	4nd	...
Ind2	2nd	1st	...
...	...	...	...

Fig. 2. Structuring data according to individual preferences, following the *Degree of Representativeness*.

$$\begin{aligned} DoR : \Omega &\rightarrow \mathbb{R}^+ \\ x &\mapsto Aggreg_{\forall y \in \Omega}(Sc_{yD}(x)) \end{aligned} \tag{2}$$

In this paper, the aggregation function is the *sum* function.
Let us define k as a scale factor allowing control over the number of exemplars. Using the k-neighborhood based on Euclidian distance, each object is linked with the neighbor having the highest DoR. With $N_k(x)$ the set of k nearest objects of x, the links are defined by :

$$\forall x \in \Omega, \quad x \mapsto y = argmax_{z \in N_k(x)} DoR(z) \tag{3}$$

Each object is linked to the most representative object in its neighborhood, and the scale factor k defines the size of the considered neighborhood, thus creating a few very representative exemplars (with a high value for k) or a lot of exemplars that closely match objects they represent (with a low value for k).

To illustrate the aforementioned algorithm, we can use *Iris*, a well known dataset. Fig. 3 (a) shows the structuring of the dataset for a scale factor of $k = 5$, resulting in many exemplars. Each exemplar is very similar to the individuals it represents : this is the configuration that will be used in a recommendation system. Like we can see on Fig. 3 (b), the number of exemplars decreases as the scale factor increases ($k = 15$). This configuration is useful to summarize a population with only a handful of key individuals, and proves to be an asset when exploring an unknown dataset. For a given scale factor, some nodes link to themselves as an exemplar : they become fewer as the scale factor is increased, eventually leaving only one node linking to itself for a scale factor equal to the size of population ($k = n$). While increasing the scale factor from 1 to n, the more iterations an individual links to itself, the more atypical it is. When applied to the CARéDIAB dataset in the context of a recommendation system, our method gives the structure illustrated on Fig. 4. To obtain this result, for each patient we extracted data regarding medical complications related to diabetes and a time series representing the evolution of glycated hemoglobin levels (HbA_1c). We then created ten quantitative indicators from this time series : the resulting dataset contains 256 individuals and 10 features.

4. Application to a visualization tool

We designed a prototype tool implementing the steps described in Fig. 1. We are currently able to automate both the filtering and structuring of a dataset, leading to an effective medical recommendation system. Our next milestone is to gather user feedback for each proposed recommendation and automatically use it to refine the similarity graph, resulting in more appropriate and customizable matching between entities. The final prototype will then be deployed for testing in a clinical environment. Once used by doctors and experts, usage of the tool and quality of the results will be surveyed in order to improve our algorithm and optimize the prototype.

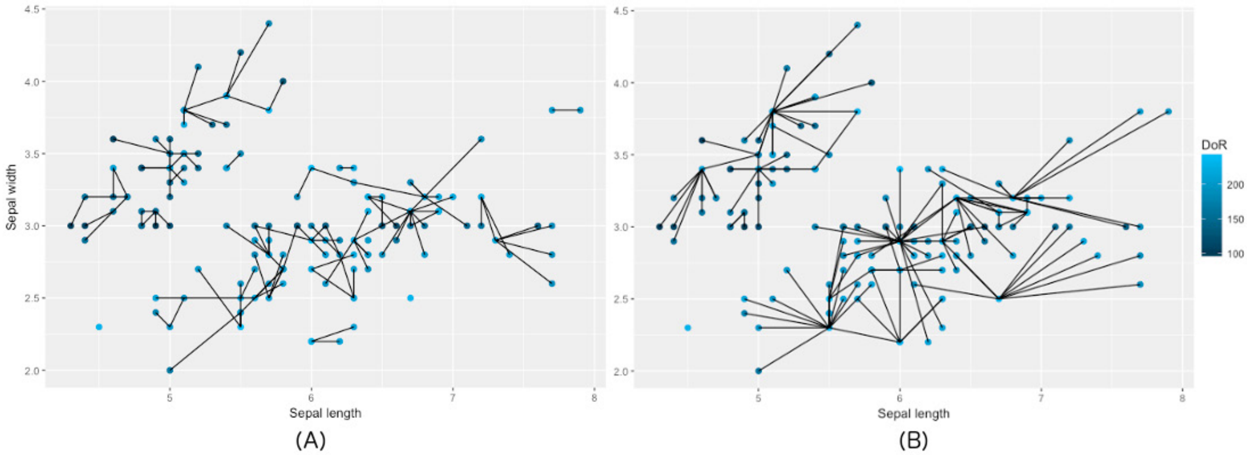


Fig. 3. Application to Iris dataset : (a) scale factor of 5, giving 49 exemplars ; (b) scale factor of 15, giving 22 exemplars.

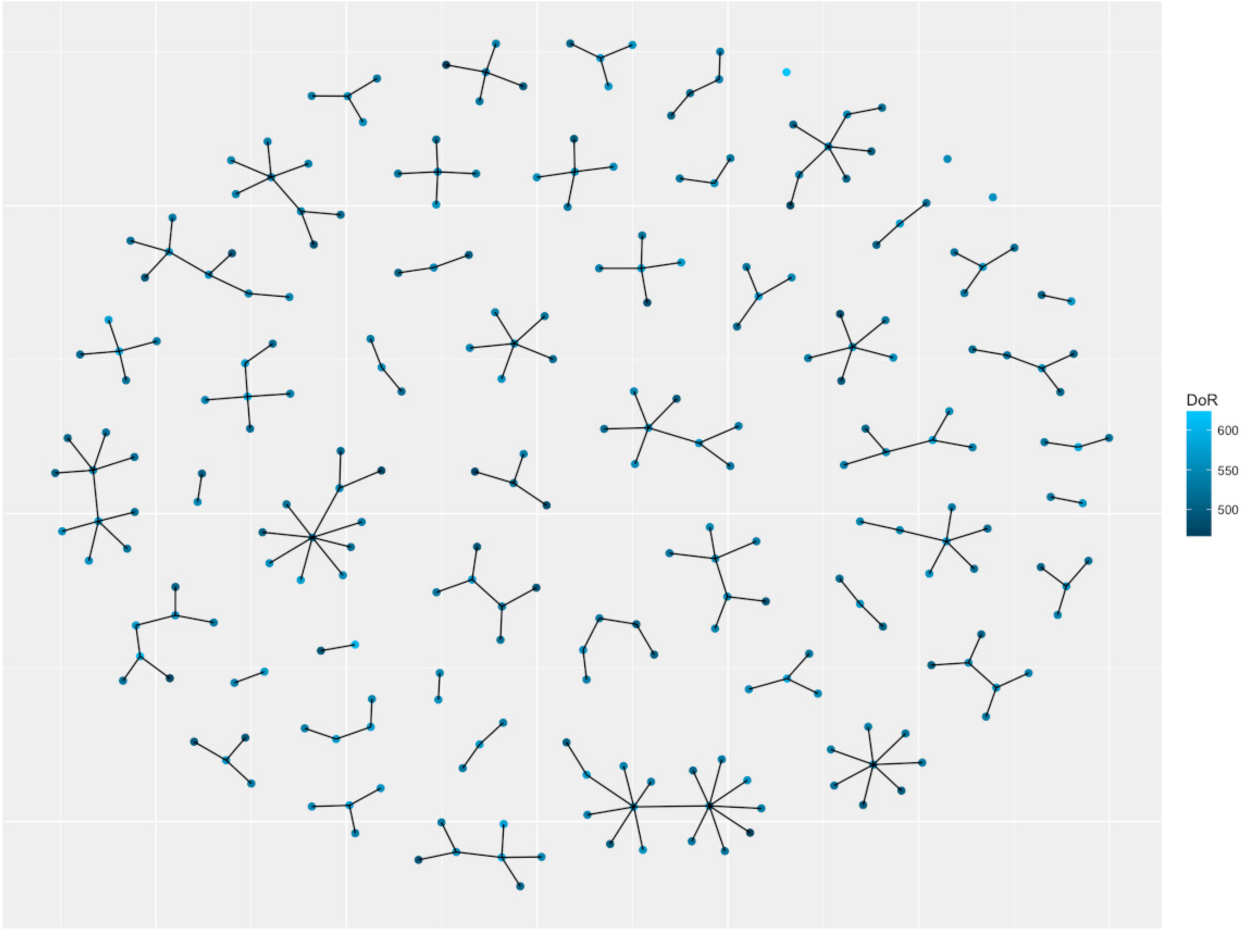
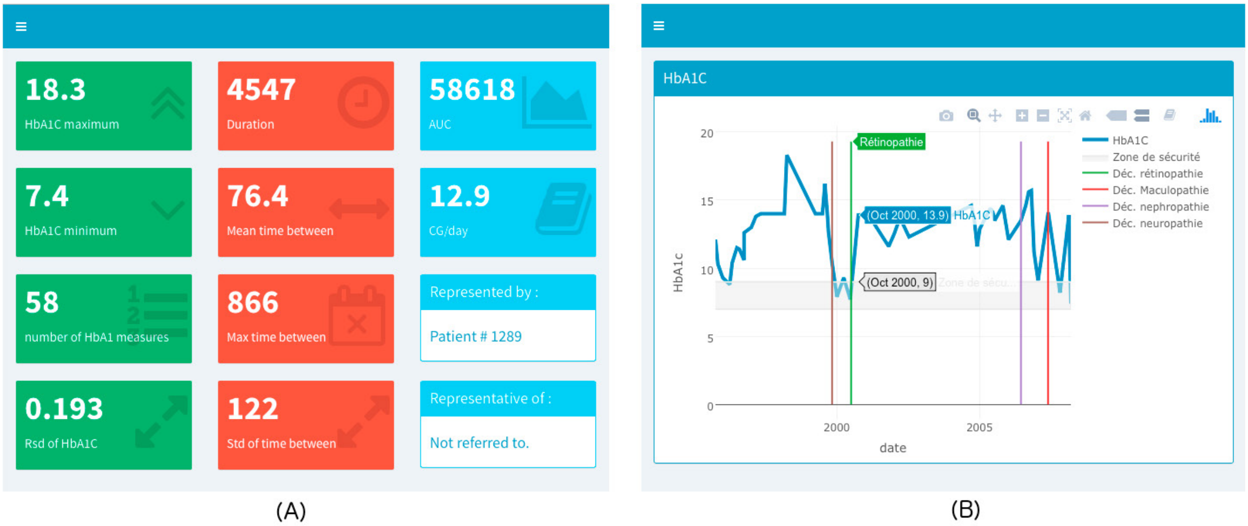


Fig. 4. Association graph representing the CARéDIAB dataset : scale factor of 5, giving 77 exemplars.

4.1. Technical choices

The prototype tool was built from the ground up using R[10]. Having no license restrictions and being platform agnostic, R imposed itself as the language of choice to test our recommendation algorithm and easily distribute it.



18.3 HbA1C maximum

4547 Duration

58618 AUC

7.4 HbA1C minimum

76.4 Mean time between

12.9 CG/day

58 number of HbA1 measures

866 Max time between

Represented by : Patient # 1289

0.193 Rsd of HbA1C

122 Std of time between

Representative of : Not referred to.

Fig. 5. (a) Patient summary; (b) Evolution of blood glucose over ten years.

The vast R ecosystem provides a handful of data management and visualization packages used for data filtering and rendering. We used Rstudio's *Shiny*[11] and *Shinydashboard*[12] packages to turn our early prototype into a full-fledged and modern web application. This ensures that we can easily deploy our recommendation system on a distant server and let medical experts use it with only a web browser and no required installation. Doctors can also use it on smartphone or tablet when visiting patients, providing a unique solution in any context. Being able to host the server ourselves and letting medical experts rely on a web client gives us the opportunity to update the tool and adapt the structuring algorithm with minimal downtimes. In addition to facilitating maintenance and deployment of new versions, logs can be collected and analyzed to review the relevance of each recommendation, and wether users deemed it appropriate or not. If needed, the prototype can be deployed on any other server using Docker. Keeping track of multiple versions and upgrading a server is thus simple, the process only requiring a new Docker image.

4.2. *Interface and user interaction*

The tool's interface is divided in three distinct parts. The first one focuses on management of the dataset. It is used to upload a dataset containing medical records and filter individuals based on the desired values. Doctors can restrict the population to match arbitrary criteria, ensuring that the working dataset is relevant to the patient currently being investigated. A table offers a summary of the filtered dataset, along with the possibility to export it for later use.

Once the filtering step is over, users can access detailed information on each patient. In the case of the CARéDIAB dataset, a dashboard sums up various indicators regarding blood glucose levels and diabetes follow-up as seen in Fig. 5 (a). As illustrated in Fig. 5 (b), glycated hemoglobin level is also available through a timeline that includes any diabetes-related complications the patient suffered.

The main focus of this prototype is to enrich data to create recommendations for medical experts. The user can change, at any time, the size of the neighborhood considered during the structuring step, dynamically adapting recommendations toward patients closely-related to the currently examined patient, or more general exemplars. Once the neighborhood criterion is set, the patient summary will include the representative of this individual according to our structuring algorithm. It will also include any other individuals that elected the current patient as their representative. A recommendation graph (Fig. 6) display the output structure of our algorithm as a graph, where each patient is represented by a node, and each node is linked to its representative individual. Using this graph, it becomes easy to browse several related medical records, inspecting any patient the user deems noteworthy.

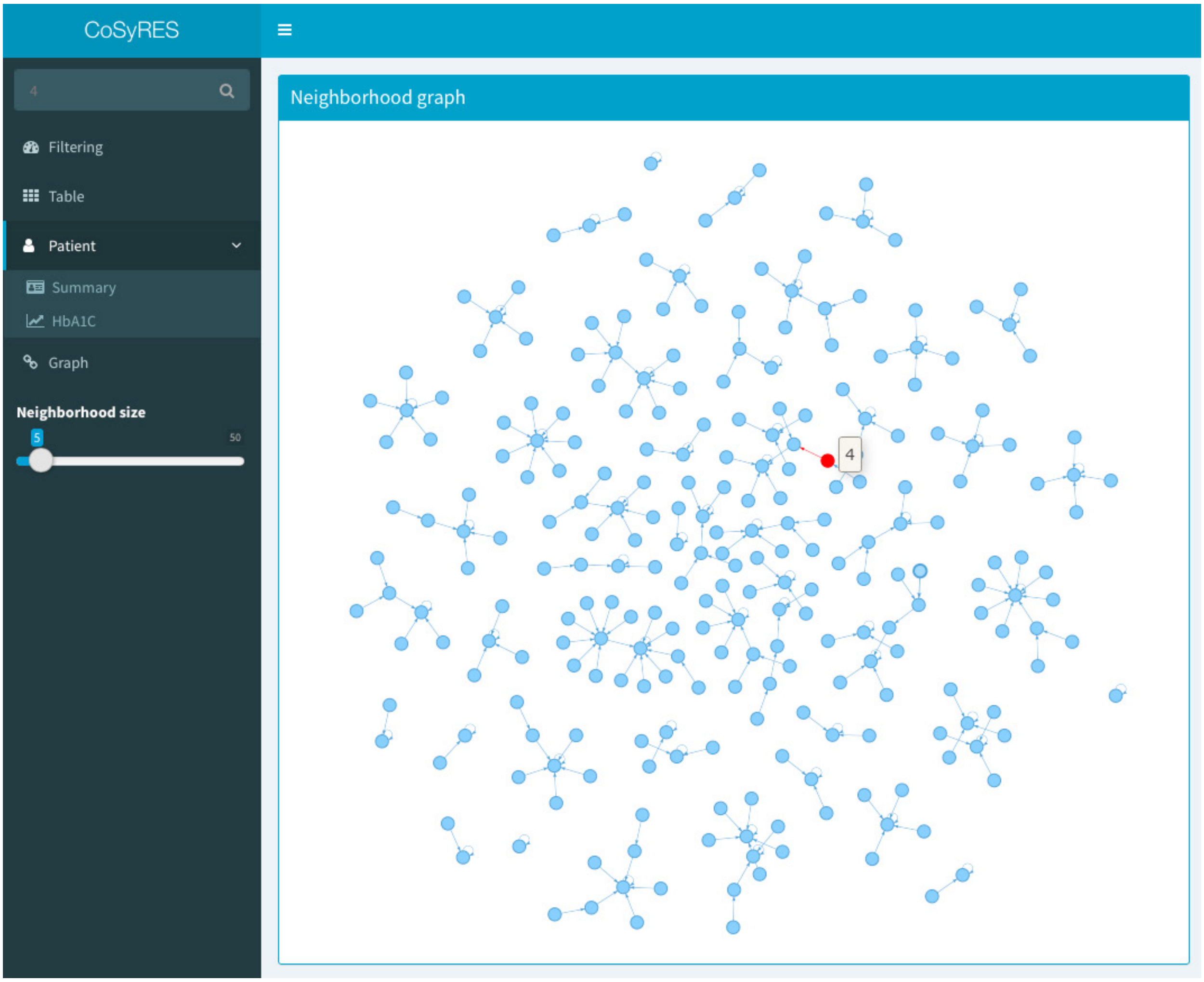


Fig. 6. Recommendation graph created with our algorithm : each patient is linked to its representative.

5. Conclusion and future works

Social choice theory can be used as an effective way to enrich data, in order to create a recommendation system. The algorithm used to structure patients according to their similarity is simple and effective, in addition to being scalable. It provides a transparent and understandable framework for medical experts to manipulate and explore medical records, and proves to be a customizable tool : each user can emphasize specific features and similarity criteria that should be favored during the structuring step. We presented a first prototype that is currently being tested and reviewed by doctors. This web application use the aforementioned algorithm as a way to link closely-related patients together, suggesting other similar medical records to the user, thus creating a framework to manipulate and explore large datasets with thousands of individuals. Future works will consider a way to automatically gather user feedback regarding recommendations to gradually improve the weighting of features, refining the recommendations as the user manipulate the dataset. To create a seamless experience, users will be solicited in an unobtrusive way using "like" and "dislike" buttons for each recommendation. Their feedback will be used as an input for the algorithm to adapt the influence of each feature during the structuring step.

References

1. Donoho, D.L., et al. High-dimensional data analysis: The curses and blessings of dimensionality. *AMS Math Challenges Lecture* 2000;**1**:32.
2. Pazzani, M., Billsus, D.. Content-based recommendation systems. *The adaptive web* 2007;:325–341.
3. Wiesner, M., Pfeifer, D.. Health recommender systems: concepts, requirements, technical basics and challenges. *International journal of environmental research and public health* 2014;**11**(3):2580–2607.
4. Obermeyer, Z., Emanuel, E.J.. Predicting the future — big data, machine learning, and clinical medicine. *New England Journal of Medicine* 2016;**375**(13):1216–1219. doi:10.1056/NEJMp1606181; pMID: 27682033.
5. Domingos, P.. A few useful things to know about machine learning. *Communications of the ACM* 2012;**55**(10):78–87.
6. Lou, Y., Caruana, R., Gehrke, J., Hooker, G.. Accurate intelligible models with pairwise interactions. In: *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*; KDD '13. New York, NY, USA: ACM; 2013, p. 623–631.
7. Lipton, Z.C.. The mythos of model interpretability. *CoRR* 2016;URL: `http://arxiv.org/abs/1606.03490`.
8. Blanchard, F., Vautrot, P., Akdag, H., Herbin, M.. Data representativeness based on fuzzy set theory. *Journal of Uncertain Systems* 2010; **4**(3):216–228.
9. Lesot, M.J., Mouillet, L., Bouchon-Meunier, B.. Fuzzy prototypes based on typicality degrees. In: *Computational Intelligence, Theory and Applications*. Springer; 2005, p. 125–138.
10. R Core Team, . *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing; Vienna, Austria; 2017.
11. Chang, W., Cheng, J., Allaire, J., Xie, Y., McPherson, J.. *shiny: Web Application Framework for R*; 2017. R package version 1.0.0.9001.
12. Chang, W.. *shinydashboard: Create Dashboards with 'Shiny'*; 2016. R package version 0.5.3.