Instituto de Ciências Matemáticas e de Computação

ISSN - 0103-2569

Similarity-based network models and how evaluate them

Robson Motta
Alneu de Andrade Lopes
Maria Cristina Ferreira de Oliveira

Nº 384

RELATÓRIOS TÉCNICOS DO ICMC

São Carlos
September/2012

Similarity-based network models and how evaluate them

Robson Motta[1]
Alneu de Andrade Lopes[2]
Maria Cristina Ferreira de Oliveira[1]

[1]University of Sao Paulo – USP
Institute of Mathematics and Computer Science – ICMC
Visualization, Imaging and Computer Graphics – VICG
P.O. Box 668
13560-970 - Sao Carlos – SP – Brasil

[2]University of Sao Paulo – USP
Institute of Mathematics and Computer Science – ICMC
Artificial Intelligence Laboratory – LABIC
P.O. Box 668
13560-970 - Sao Carlos – SP – Brasil

rmotta@icmc.usp.br,alneu@icmc.usp.br,cristina@icmc.usp.br

Abstract. *Similarity-based network models have been used in many data mining tasks, such as classification and clustering. These models are applied in non-relational data, where each example is represented by a vector of characteristics, creating a relational representation based on similarity among the examples. Using this representation, it is possible to use complex network measures, in a relational data mining context, incorporating more information than the traditional propositional data mining. In this technical report we present a new network model based on similarity, the Extended Minimum Spanning Tree network (EMST), and three measures to evaluate the network models, based on neighborhood, clusters and outliers preservation. The model proposed here is non-parametric and present good results for all evaluation measures when compared with the other models.*

Contents

1 Introduction 1

2 Similarity-based network models 2

2.1 $kNNG$ 2

2.2 MST 2

2.3 $PKNNG$ 3

2.4 kDR 4

2.5 ϵNN 5

2.6 DHN 6

3 Extended Minimum Spanning Tree (*EMST*) network model 7

4 Proposed measures to network evaluation 9

4.1 Neighborhood Quality 9

4.2 Clustering Structure Quality 10

4.3 Outliers Structure Quality 11

5 Empirical evaluation 11

6 Conclusion 12

References 16

1. Introduction

Similarity-based network models allow to create a relational representation for a non-relational dataset. A relational representation allows to explore network measures in mining tasks, in addition to the traditional measures.

In a similarity-based network the vertices represent data instances and the edges connect pairs of instances that are highly similar. Several similarity network models can be considered to obtain a network. The adoption of different strategies to create a network implicates in having different specific characteristics. Some models enforce a minimum number of neighbors on all vertices, such as in the network based on K neighbors. In these models, high values of K result in many edges inserted even among isolated examples. On the other hand, low values result in placing few connections in dense data regions. Other models present a better connection distribution, with few connections linking isolated examples and many connections linking dense data regions. These models seek to optimize the network's modular structure, even though this may not be a characteristic of the dataset. Moreover, most of the network models are parametric, making it difficult to properly set an ideal parameter for each dataset.

Looking to obtain a similarity-based network model that preserves the properties of the original data without the strong parametrical dependence, in this report we propose a non-parametrical network model, the Extended Minimum Spanning Tree network ($EMST$). This proposed network model is based on the minimum spanning tree to a complete graph, and new edges are inserted for each example considering its region in the original data space, that is, dense region has more edges inserted and sparse region less edges.

In this report we compare the $EMST$ network against some of the existing models. We proposed three measures to evaluate similarity-based network models, considering the local connections, outliers structure and cluster structure preservation.

In the Section 2 some network models are presented. In Section 3 it is presented the proposed network model. In the Section 4 it is presented a way to evaluate the network models, with three proposed measures, and in the Section 5 the empirical evaluation is shown.

2. Similarity-based network models

Network models based on similarity are used in different data mining tasks, such as classification, identification of clusters and search algorithms. The models used in this report are: k-nearest neighbors graph ($kNNG$) (Paredes et al., 2006), minimum spanning tree graph (MST) (Zhong et al., 2011), Penalized K-Nearest-Neighbor-Graphs ($PKNNG$) (Bayá and Granitto, 2011), k degree-reduced nearest neighbor graph (kDR) (Aoyama et al., 2011), ϵ-nearest neighbors graph (ϵNN) (Bayardo et al., 2007; Xiao et al., 2008; Vernica et al., 2010), and deterministic hierarquical network(DHN) (Motta et al., 2008). In addition to these models, we propose here a network model based on MST graph called $EMST$ (Extended Minimum Spanning Tree).

2.1. $kNNG$

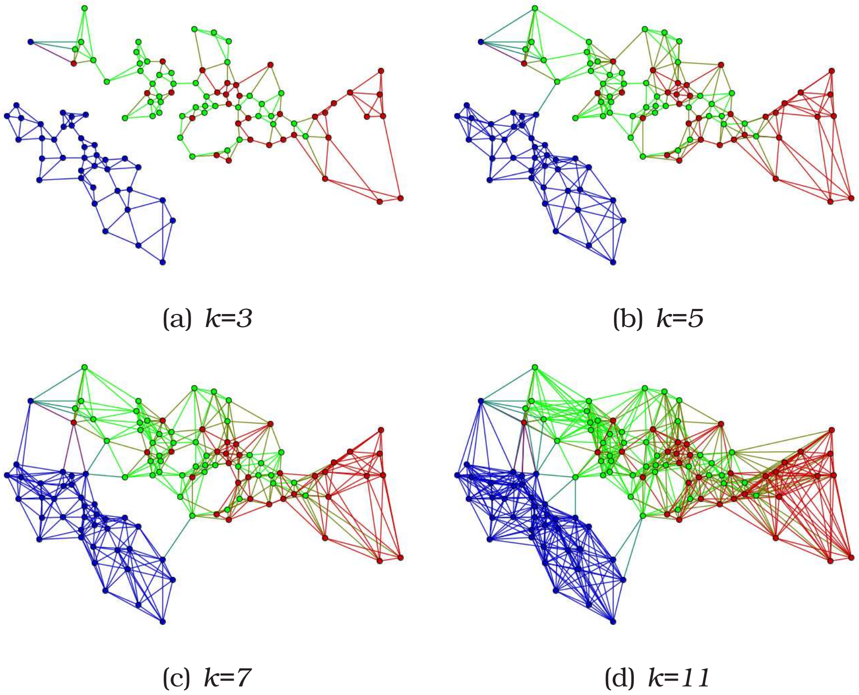
The k-nearest neighbors graph ($kNNG$) (Paredes et al., 2006) is a traditional model. A $kNNG$ from a set of data instances is obtained by connecting each instance to its k-nearest neighbors. Varying parameter k will produce networks with different properties. Graph connectivity is usually required to compute most of the network metrics, particularly those that find paths in the graph. Typically, this network model needs a high value for k to obtain a connected $kNNG$.

To illustrate the network models, in this report we consider an artificial dataset with three classes (mapped by the point colors). This is a bi-dimensional dataset, which turns possible to plot the dataset directly. The $kNNG$ is shown in the Figure 1, with k values equal to 3, 5, 7 and 11.

2.2. MST

Such as the $kNNG$, the minimum spanning tree graph (MST) is also a traditional model. Recently Zhong et al. (2011) proposed a novel split-and-merge hierarchical clustering method using a MST graph.

The MST of a graph is a sub-graph (a tree) of the original graph that includes all its vertices and has the minimum number of edges. For weighted graphs, the MST has the edge set with minimum total cost. An MST may be computed from the complete weighted graph formed by connecting each data instance to all the others, taking as edge weights their pairwise distances.



(a) *k=3*

(b) *k=5*

(c) *k=7*

(d) *k=11*

Figure 1. $kNNG$ **for an artificial bidimensional dataset using** k **equals to 3, 5, 7 and 11.**

The MST graph is a non-parametrical model, but the minimum total cost characteristic naturally creates a network with few edges and usually not preserving important properties of the data (Figure 2).

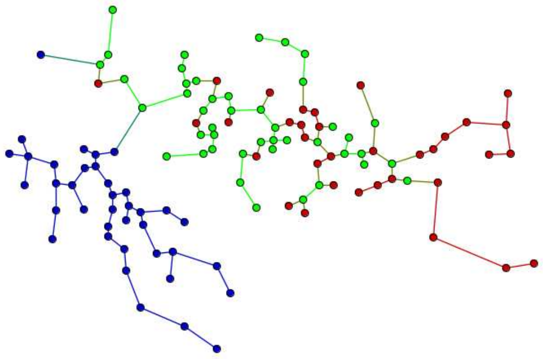


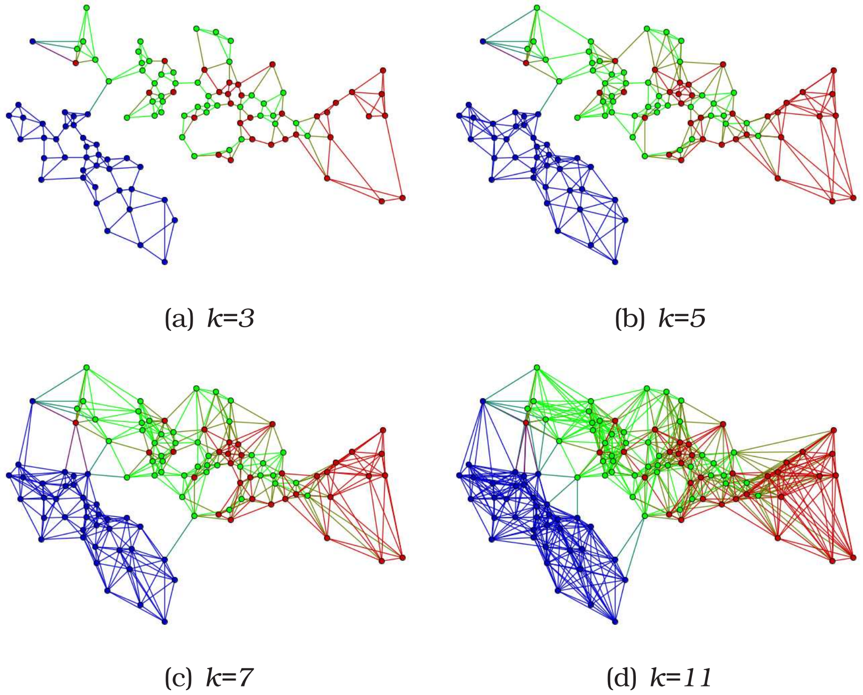
Figure 2. MST **for an artificial bidimensional dataset.**

2.3. $PKNNG$

Bayá and Granitto (2011) have build similarity-based network models to identify clusters. Connectedness is an essential property of those networks, ensuring that a finite distance path exists between any pair of instances. Networks are constructed departing from a KNN graph, which is transformed into a connected graph – these networks are called *Penalized K-Nearest-Neighbor-Graphs (PKNNG)*. The rationale is that edges linking nodes in different components are assigned weights significantly lower than those edges which are internal to a component.

Four different strategies are considered to obtain a connected graph: (i) adding the set of minimum-weighted edges required to make the KNN graph connected, i.e., the edges in its minimal spanning tree; (ii) adding the lowest-cost edge between all pairs of components; (iii) adding edges with penalized weight between all instances from distinct components; and (iv) finding the medoids of each component, to connect each pair of instances considered central. The first strategy presented the best results for cluster identification, with performance similar to state-of-the-art clustering methods.

In the Figure 3 the first model, combining KNN and MST networks, is shown with different values to k.


(a) k=3

(b) k=5

(c) k=7

(d) k=11

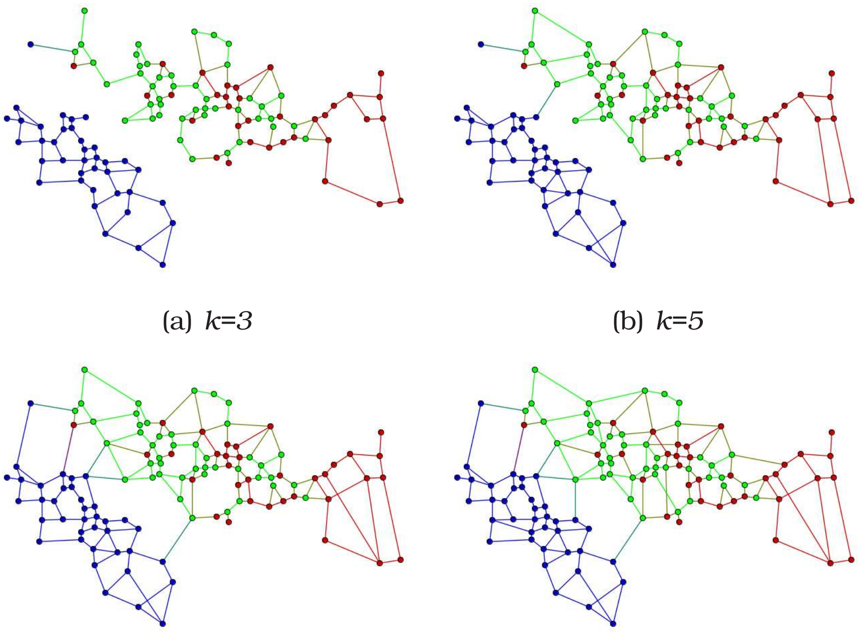
Figure 3. $PKNNG$ **for an artificial bidimensional dataset using** k **equals to 3, 5, 7 and 11.**

2.4. kDR

The k degree-reduced nearest neighbor graph (kDR) was proposed by Aoyama et al. (2011) to do a fast approximate similarity search method.

The kDR differs from kNN in not having an edge between x and $y \in N_k(x)$ without which a greedy search algorithm can reach x from y along the existing edges. Then kDR has a smaller average degree than kNN.

To construct the kDR graph, a kNN network is created with $k = 1$. After this, it is adopted an incremental procedure on k, until k reaches $k = k_{max}$, where k_{max} is provided in advance. The kDR graph starts exactly a kNN graph with k equals to 1. Then, for each kNN graph, the edges are verified and may be inserted in the kDR graph. An edge from x to y is inserted if in a current kDR graph the distance between x and every adjacent to y is bigger then the distance between x and y. The final network is close to $kNNG$ to low k values. The results for our practical example are shown in the Figure 4.


(a) *k=3*

(b) *k=5*

(c) *k=7*

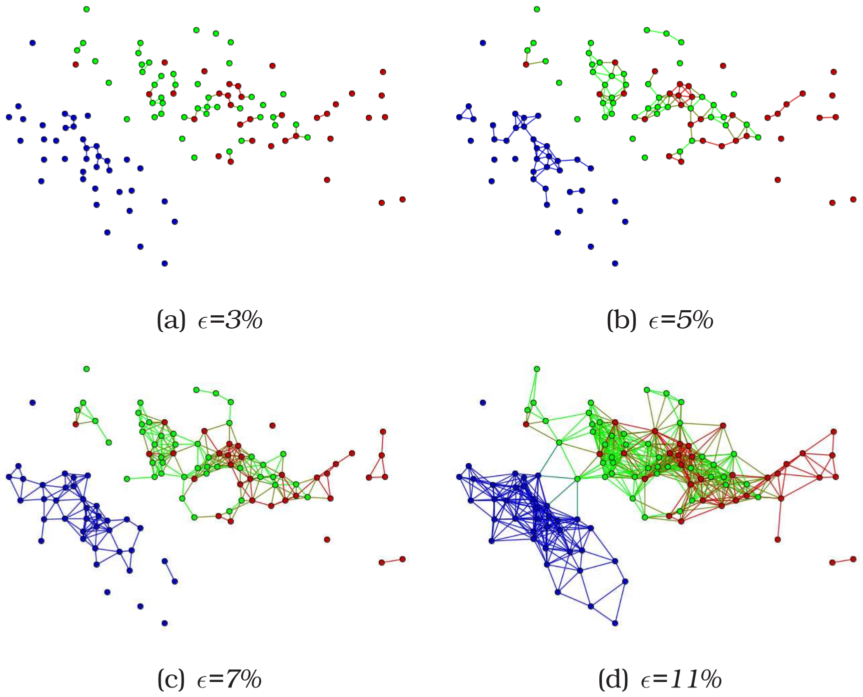
(d) *k=11*

Figure 4. kDR **for an artificial bidimensional dataset using** k **equals to 3, 5, 7 and 11.**

2.5. ϵNN

There is some work related to efficient and fast algorithms to find pairs of examples with high similarity in large datasets (Bayardo et al., 2007; Xiao et al., 2008; Vernica et al., 2010). These works are related to a way to find a graph which all pairs of examples with similarity higher than the threshold are connected, which we called ϵNN.

The problem is that such methods are dependent of the threshold. It is common to create a very sparse and disconnected graph with a low threshold value, and a highly connected graph with a higher threshold value. In the Figure 5, we used, in our practical example, the threshold values equal to 3%, 5%, 7% and 11% of the higher distance among the points.


(a) ϵ=3%

(b) ϵ=5%

(c) ϵ=7%

(d) ϵ=11%

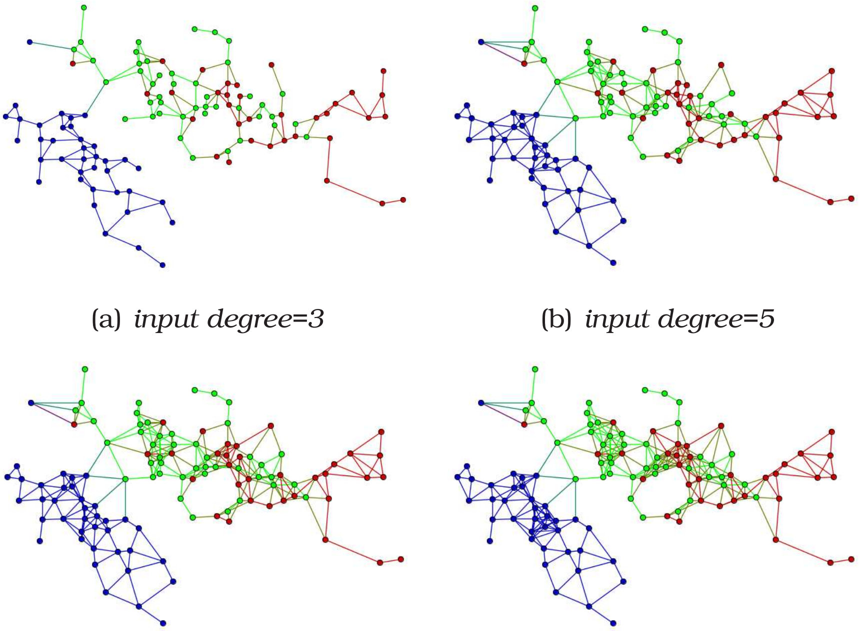
Figure 5. ϵNN **for an artificial bidimensional dataset using** ϵ **equals to 3%, 5%, 7% and 11% of the higher distance in the dataset.**

2.6. *DHN*

Motta et al. (2008) proposed a *Deterministic Hierarchical Network model (DHN)* for community detection in complex networks. The goal is to obtain a modular network, formed by connected components, or communities, with few connections across communities and many connections within communities.

Given N data instances, an initial network model with N one-vertex components and no connections are built. A hierarchical agglomerative process connects vertex pairs iteratively, given a minimum similarity threshold, until reaching a user-defined average degree for the component. The coalition process greedily optimizes a component interconnectivity criterion that considers both the similarity between vertices and the number of connections across and within components.

In other words, it searches for new components that have highly similar vertices, greater number of connections intra-component and few connections to other components. The procedure stops once a connected graph is obtained, illustrated in the Figure 6.


(a) *input degree=3* (b) *input degree=5*

(c) *input degree=7* (d) *input degree=11*

Figure 6. DHN **for an artificial bidimensional dataset using input degree equals to 3, 5, 7 and 11.**

3. Extended Minimum Spanning Tree (*EMST*) network model

In this report we introduce a network model named *Extended Minimum Spanning Tree* (*EMST* network), which expands a graph's *Minimum Spanning Tree* (*MST*) by connecting each vertex to its most similar vertices employing a criterion that considers the already existing connections. Building the *EMST* network from the data comprises two stages: first a complete weighted graph is created, with edge weights given by the pairwise dissimilarity values between vertices. Departing from the graph's *MST*, which is a connected graph with minimum global weight, edges are then added based on connection patterns identified in the *MST*. The resulting network preserves the original data distribution, with vertices in dense data regions highly connected and isolated vertices sparsely connected.

Algorithm 1 describes the construction process. First, the complete graph G_C is obtained, and the edges contained in the *MST* of G_C are added to a different edge set A'. An empty edge set A is then created and the edges start being added to it. For each vertex v_i, a $limit$ measure is computed considering the maximum value between the weight to the closest vertex ($w_{a_{ij}}^{G_C}$) and the global average weight ($\langle w^{A'} \rangle$), and a value that considers $\langle w^{A'} \rangle$ and a distance distribution quality Q_{dd}.

The distance distribution quality Q_{dd} is described by Equation 1. D is the normalized distance between examples. The standard deviation sd, which has values within [0;0.5], is taken as a distance distribution quality. Its maximum value is a normalization factor. A Q_{dd} value close to zero indicates that many examples are within a small distance from a reference example, whereas a value close to one indicates the opposite.

$$Q_{dd}(D) = sd(D)/0.5 \tag{1}$$

The final network is given by $G_{EMST}(V, A)$, where A contains A'.

Algorithm 1 *Extended Minimum Spanning Tree network model* (*EMST*)

```
Input:
  Set of examples: $X$ = $x_1$,...,$x_n$
  Distance function: $d$
Output:
  EMST network: ($V$,$A$)

Vertices $V \leftarrow X$
Distance matrix $D$ (pairwise dissimilarities) $\leftarrow d(X)$ // normalized
$G_C \leftarrow$ CompleteGraph($V, D$) // complete weighted (by distances) graph
Edges $A' \leftarrow MST(G_C)$ // edges from MST($G_C$)
Edges $A \leftarrow$ Ø
For each vertex $v_i$ in $V$
  $v_j \leftarrow$ closest vertice to $v_i$ in $G_C$
  $limit \leftarrow max\_arg(w_{a_{ij}}^{G_C}, \langle w^{A'} \rangle) + (\langle w^{A'} \rangle * Q_{dd}(D))$
  For each vertex $v_k$ in $V$
    $A \leftarrow A \cup \{a_{ik} \mid a_{ik} \in G_C$ and $w_{a_{ik}}^{G_C} \leq limit\}$

Returns ($V$,$A$)
```

Ideally, an *EMST* network has densely connected communities of highly similar examples, mostly from the same class, and a few edges linking vertices in different communities. In other words, the network's community structure and topology are expected to reflect the underlying similarity structure of the data. The *EMST* algorithm is $O(n^2)$, where n is the number of vertices.

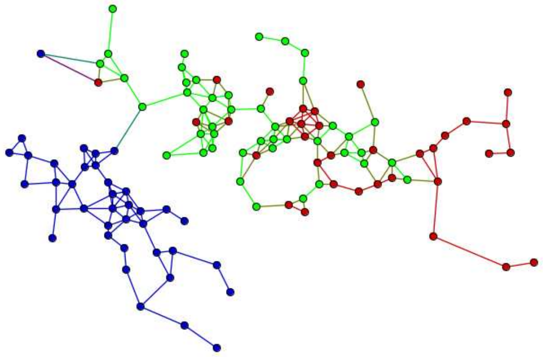


Figure 7. $EMST$ for an artificial bidimensional dataset.

4. Proposed measures to network evaluation

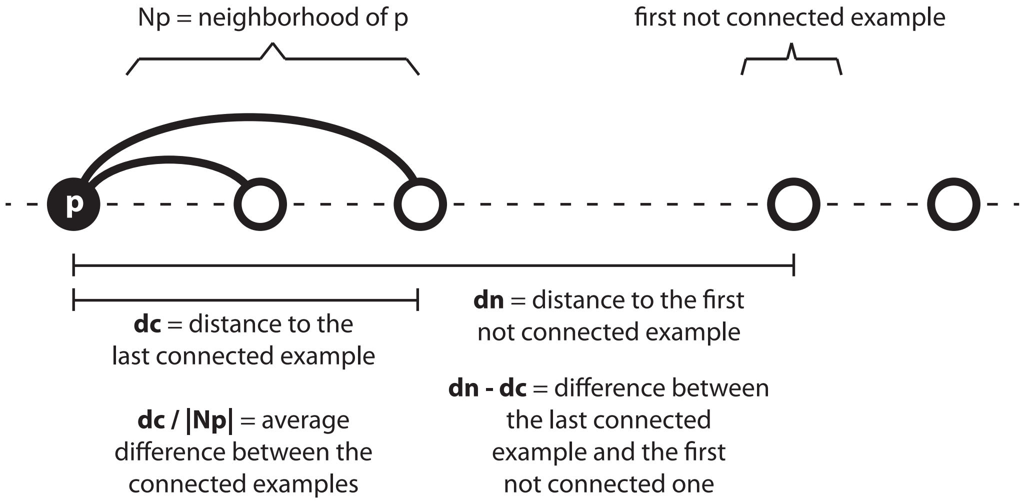
In this section, we present the measures that we used in this paper to evaluate and compare the similarity-based networks. All these measures were intended to measure different aspects from these networks: neighborhood quality, clustering structure and outliers structure. All these measures are detailed in the next subsections.

4.1. Neighborhood Quality

In order to evaluate the neighborhood quality of an example we can consider its neighborhood (adjacent examples) and the closest not connected example. A good neighborhood is one with small distance between the connected examples and a big distance between the last connected example and the first not connected example, as presented in the Figure 8.

In Figure 8 it is shown a reference point p with two connected examples (N_p set), a distance d_c between p and its last connected example and a distance d_n between p and its first not connected example. Then, $\frac{d_c}{|N_p|}$ is the average difference between the connected examples, and $d_n - d_c$ is the difference between the last connected example and the first not connected example. A good neighborhood quality to an example p is a high value to $d_n - d_c$ and low value to $\frac{d_c}{|N_p|}$, described in Equation 2. For all examples in a network, the Equation 3 describe the neighborhood quality.

$$q_{neighborhood}(p) = \frac{(d_n - d_c)}{(d_n - d_c) + (\frac{d_c}{|N_p|})} \tag{2}$$



Np = neighborhood of p
first not connected example
p
dc = distance to the last connected example
dn = distance to the first not connected example
dc / |Np| = average difference between the connected examples
dn - dc = difference between the last connected example and the first not connected one

Figure 8. Example of a neighborhood in a 1-dimensional dataset.

$$Q_{neighborhood}(R) = \frac{1}{|R|}\sum_{p\in R} q_{neighborhood}(p) \tag{3}$$

4.2. Clustering Structure Quality

Analyzing clusters is a common task in visual data mining. Therefore, it is important that the network preserves the clustering structure, that is, a network with many edges among examples of the same cluster and few edges among examples in different clusters.

First of all, to evaluate the clustering structure quality, it is important identify the clusters in the dataset using some clustering algorithm. A simple algorithm is the X-Means (Dan Pelleg, 2000), which is a extension of the K-Means (Duda and Hart, 1973) using only two parameters, lower and higher value to k instead of the strong parameter k in K-Means. Basically, the X-Means algorithm increases the parameter K, between the lower and higher input values, and uses a quality measure to find the better cluster structure.

The second step is to create a quality measure with the edges among examples in different clusters. The proposed measure is described in Equation 4, where E is the set of all edges and E_i is the set of edges intra-clusters, that is, edges among example of the same cluster.

$$Q_{cluster}(R) = \frac{|E_i|}{|E|} \tag{4}$$

Naturally the $Q_{cluster}$ measure will create high values for most of the network models, but here we are considering just a comparison among the models and not a value directly related to the quality of the cluster structure.

4.3. Outliers Structure Quality

The concept of outliers structure quality is similar to clustering structure quality. However, in this case, the outliers should have few adjacent examples comparing to the not outliers in the network.

Similar to the clustering, first of all it is important to identify the outliers in the dataset. There are many algorithms to identifyu identify outliers (Knorr and Ng, 1997; Ramaswamy et al., 2000; Schölkopf et al., 2001; Ester et al., 1996). Here we consider the Local Outlier Factor - LOF (Breunig et al., 2000), which creates a measure that quantifies how much each example is an outlier, called outlier factor. Using this measure, it is created a ranking of the examples to be analysed. With this ranking, it is possible to select the outliers in each dataset.

Once the outliers are detected, we need to create an outliers quality measure to apply in the networks. We propose here a measure to compare the average degree of the outliers (number of adjacent examples) with the average degree of the network. The Equation 5 describes the proposed measure, where $\langle d \rangle$ is the average degree of the network and $\langle d_o \rangle$ is the average degree of the outliers. The min condition is used because the $\langle d_o \rangle$ can be higher then $\langle d \rangle$, which is the worst case.

$$Q_{outlier}(R) = 1 - min(1, \frac{\langle d \rangle}{\langle d_o \rangle}) \tag{5}$$

5. Empirical evaluation

In order to compare the efficiency of the different network models, we used 12 numeric datasets from the UCI repository (balance, cleveland, diabetes, ecoli, heart-statlog, ionosphere, iris, satimg, sonar, vehicle, vowels and wine), 2 image datasets (corel and medical) and 2 textual datasets (CBR-ILP-IR-SON and KDViz). Table 1 shows a brief summary about these datasets.

We applied four different similarity-based network algorithms to these datasets: kNN network model with k equals to 3, 5, 7 and 11;

Table 1. Data sets for experimental evaluation.

dataset	# examples	# attributes	# clusters	# outliers
balance	625	4	2	0
cleveland	298	13	3	5
diabetes	768	8	2	6
ecoli	336	7	3	1
heart-statlog	270	13	2	5
ionosphere	351	34	3	2
iris	150	4	2	3
satimg	500	36	4	4
sonar	208	60	4	3
vehicle	846	18	4	9
vowels	990	10	2	2
wine	178	13	3	5
medical	540	28	4	6
corel	1000	150	4	1
CBR-ILP-IR-SON	675	1783	3	3
KDViz	1624	520	3	2

minspan network model with k equals to 3, 5, 7 and 11; minimum spanning tree network (MST); extended minimum spanning tree network ($EMST$).

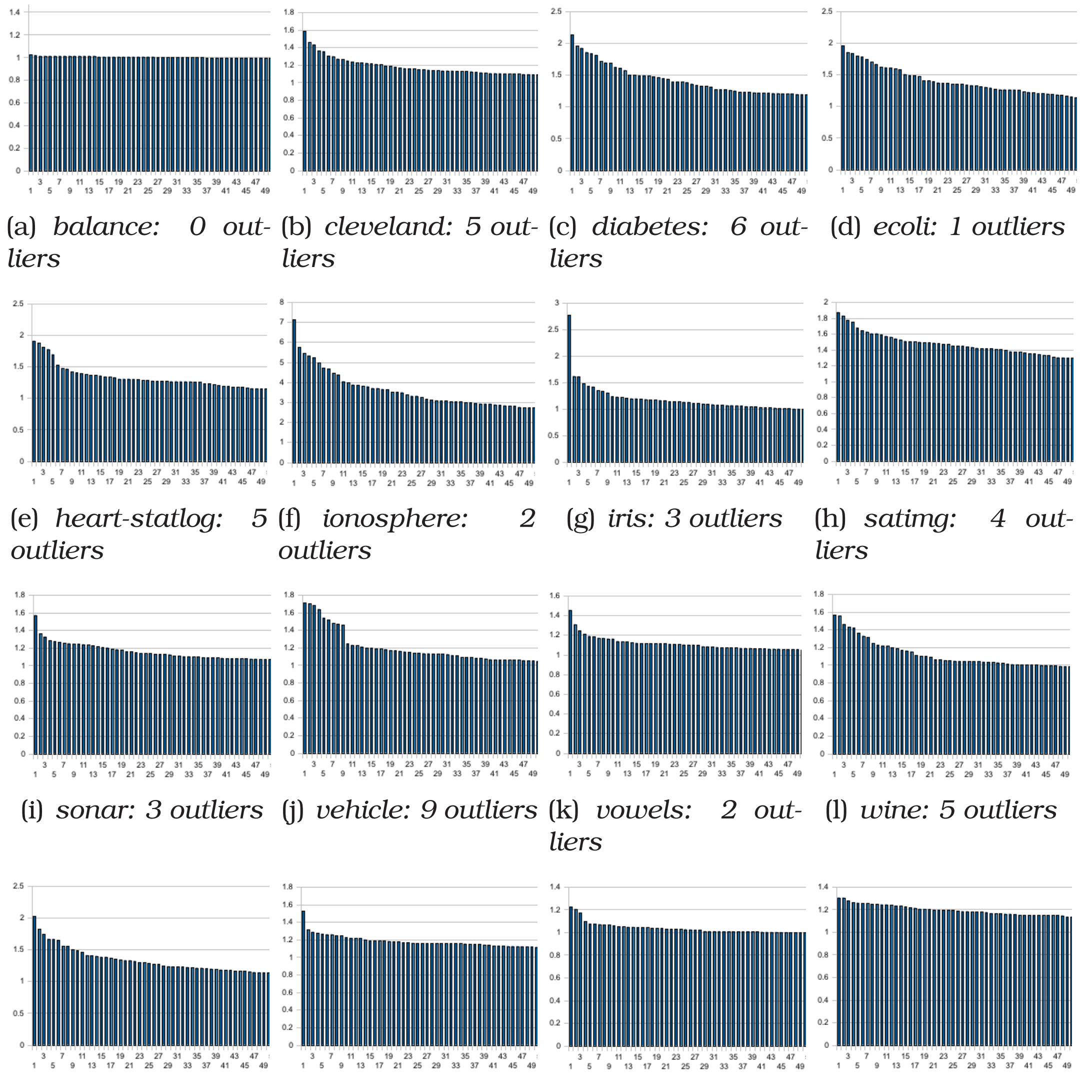
In order to identify clusters, it was used the X-Means with minimum number of clusters equals to 2 and maximum equals to 50. Also, to identify outliers, it was used the Local Factor Outlier algorithm to create a ranking, and the outliers were defined after we observe the outlier factor values in the ranking (Figure 9).

Table 2 contains, for each network model, the percent of examples of the dataset that were connected in the network, the average degree $\langle d \rangle$, the average values for the three network quality proposed measures, and the average of these three values, with the final ranking. Figure 10 shows the values of each one of the proposed measures to network quality.

6. Conclusion

In this report we proposed a new non-parametric network model ($EMST$ network) and three measures to be used in comparison of network models: evaluation the preservation of the data properties, such as neighborhood, cluster and outlier structures. In the comparison, we used some similarity-based network models.

The networks were applied and evaluated over 12 numeric datasets, 2 image datasets and 2 textual datasets. The ground truth consider to the clusters and outliers were obtained with, respectively,



(a) *balance: 0 outliers* (b) *cleveland: 5 outliers* (c) *diabetes: 6 outliers* (d) *ecoli: 1 outliers*

(e) *heart-statlog: 5 outliers* (f) *ionosphere: 2 outliers* (g) *iris: 3 outliers* (h) *satimg: 4 outliers*

(i) *sonar: 3 outliers* (j) *vehicle: 9 outliers* (k) *vowels: 2 outliers* (l) *wine: 5 outliers*

(m) *medical: 6 outliers* (n) *corel: 1 outliers* (o) *CBR-ILP-IR-SON: 3 outliers* (p) *KDViz: 2 outliers*

Figure 9. Ranking of the Local Factor Outlier for each dataset and how many outliers were selected in each dataset.

X-Means and Local Factor Outlier (LOF) algorithms.

Comparing the network models, the neighborhood quality of the $EMST$ and DHN networks presented the lower ranking value, indicating better results. The better results to clustering quality were for the $KNNG$ network with k equals to 3, but this network had only 50% of the connected value. And the better results to outlier quality were presented also to DHN network with degree equals to 11.

The DHN with degree equals to 11 presented good values to neighborhood and outlier quality, but one of the three worst values to cluster quality. The proposed $ESMT$ network model presented good results in

Table 2. Network information (connected and average degree) and average ranking of each proposed measure (neighborhood, clustering and outliers qualities), with the final average ranking.

network	connected	$\langle d \rangle$	neighborhood quality (rank)	clustering quality (rank)	outliers quality (rank)	average ranking
KNN3	50%	4.38	12.81	**4.5**	9.75	9.02 (7)
KNN5	68.8%	7	13.38	7.88	11.06	10.77 (11)
KNN7	81.3%	9.71	13.56	11.38	12.31	12.42 (16)
KNN11	87.5%	15.25	14.5	15	12.81	14.1 (19)
KDR3	43.8%	3.19	15.56	5.5	13.13	11.4 (14)
KDR5	68.8%	4.08	12.25	8.25	13.19	11.23 (12)
KDR7	81.3%	4.72	10.75	10.94	12.75	11.48 (15)
KDR11	87.5%	5.49	8.63	14.56	14.38	12.52 (17)
ENN connected	100%	133.64	3.63	17.88	4.38	8.63 (6)
DHN3	100%	3	7.69	5.75	7.19	6.88 (2)
DHN5	100%	4.89	5.31	10.81	4.81	6.98 (4)
DHN6	100%	6.68	4.06	12.69	4.81	7.19 (5)
DHN11	100%	9.96	**3.56**	13.38	**3.81**	6.92 (3)
MinSpan3	100%	4.42	13	5.88	10.56	9.81 (9)
MinSpan5	100%	7.01	13.63	8.63	11.63	11.29 (13)
MinSpan7	100%	9.71	13.81	11.94	12.63	12.79 (18)
MinSpan11	100%	15.25	14.5	15.5	12.81	14.27 (20)
MST	100%	1.99	15.19	5	10.19	10.13 (10)
EMST	100%	12.27	**3.56**	5.31	5	**4.63 (1)**

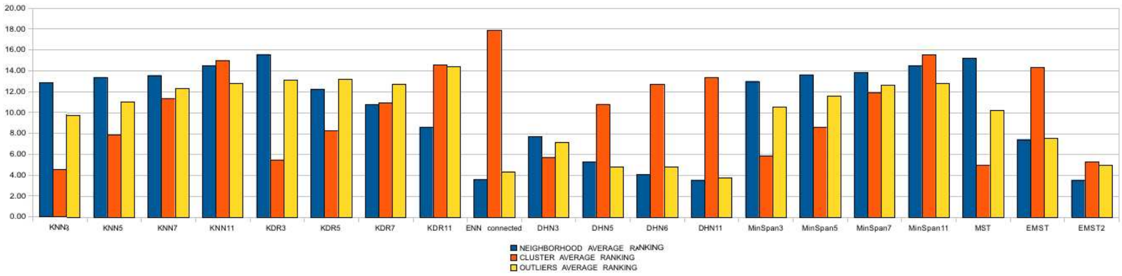


Figure 10. Average ranking of each measure: neighborhood, cluster and outlier quality.

the three proposed measures, being the best one considering the final average ranking.

Considering the measures proposed here, the $EMST$ network model preserves the properties of the dataset, having a good neighborhood, cluster and outlier structure. Furthermore, the $EMST$ network is a non-parametric model.

References

Aoyama, K., Saito, K., Sawada, H., and Ueda, N. (2011). Fast approximate similarity search based on degree-reduced neighborhood graphs. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '11, pages 1055–1063, New York, NY, USA. ACM. Cited in pages 2 and 4.

Bayá, A. E. and Granitto, P. M. (2011). Clustering gene expression data with a penalized graph-based metric. *BMC Bioinformatics*, 12:2. Cited in pages 2 and 3.

Bayardo, R. J., Ma, Y., and Srikant, R. (2007). Scaling up all pairs similarity search. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 131–140, New York, NY, USA. ACM. Cited in pages 2 and 5.

Breunig, M., Kriegel, H.-P., Ng, R. T., and Sander, J. (2000). Lof: Identifying density-based local outliers. In *PROCEEDINGS OF THE 2000 ACM SIGMOD INTERNATIONAL CONFERENCE ON MANAGEMENT OF DATA*, pages 93–104. ACM. Cited in page 11.

Dan Pelleg, A. M. (2000). X-means: Extending k-means with efficient estimation of the number of clusters. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 727–734, San Francisco. Morgan Kaufmann. Cited in page 10.

Duda, R. O. and Hart, P. E. (1973). *Pattern Classification and Scene Analysis*. John Wiley & Sons Inc. Cited in page 10.

Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, pages 226–231. Cited in page 11.

Knorr, E. M. and Ng, R. T. (1997). A unified approach for mining outliers. In *In Proc. KDD*, pages 219–222. Cited in page 11.

Motta, R., Almeida, L. J., and Lopes, A. A. (2008). Redes probabilísticas baseadas em similaridade na exploração de comunidades. In *I Workshop on Web and Text Intelligence (SBIA-WTI08)*, pages 11–18, Salvador, Brasil. Cited in pages 2 and 6.

Paredes, R., Chávez, E., Figueroa, K., and Navarro, G. (2006). Practical construction of *k*-nearest neighbor graphs in metric spaces. In *WEA*, pages 85–97. Cited in page 2.

Ramaswamy, S., Rastogi, R., and Shim, K. (2000). Efficient algorithms for mining outliers from large data sets. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, SIGMOD '00, pages 427–438, New York, NY, USA. ACM. Cited in page 11.

Schölkopf, B., Platt, J. C., Shawe-Taylor, J. C., Smola, A. J., and Williamson, R. C. (2001). Estimating the support of a high-dimensional distribution. *Neural Comput.*, 13:1443–1471. Cited in page 11.

Vernica, R., Carey, M. J., and Li, C. (2010). Efficient parallel set-similarity joins using mapreduce. In *Proceedings of the 2010 international confer-*

ence on Management of data, SIGMOD '10, pages 495–506, New York, NY, USA. ACM. Cited in pages 2 and 5.

Xiao, C., Wang, W., Lin, X., and Yu, J. X. (2008). Efficient similarity joins for near duplicate detection. In *Proceedings of the 17th international conference on World Wide Web*, WWW '08, pages 131–140, New York, NY, USA. ACM. Cited in pages 2 and 5.

Zhong, C., Miao, D., and Fränti, P. (2011). Minimum spanning tree based split-and-merge: A hierarchical clustering method. *Information Sciences*, 181(16):3397–3410. Cited in page 2.