Informativity-Based Graph: Exploring Mutual kNN and Labeled Vertices for Semi-Supervised Learning

Lilian Berton, Alneu de Andrade Lopes
Instituto de Ciências Matematicas e de Computacão
Universidade de São Paulo - Campus de São Carlos, CP 668
13560-970 São Carlos, SP - Brazil
Email: lberton, alneu@icmc.usp.br

***Abstract*—Data repositories are getting larger and in most of the cases, only a small subset of their data items is labeled. In such scenario semi-supervised learning (SSL) techniques have become very relevant. Among these algorithms, those based on graphs have gained prominence in the area. An important step in graph-based SSL methods is the conversion of tabular data into a weighted graph. However, most of the SSL literature focuses on developing label inference algorithms without studying graph construction methods and its effect on the base algorithm performance. This paper provides a novel technique for building graph by using mutual kNN and labeled vertices. The use of prior information, i.e., to consider the small fraction of labeled vertices, has been underexplored in SSL literature and mutual kNN has been only explored in clustering. The empirical evaluation of the proposed graph showed promising results in terms of accuracy, when it is applied to the label propagation task. Additionally, the resultant networks have lower average degree than kNN networks.**

***Keywords*-graph construction; classification; semi-supervised learning**

I. Introduction

Labeling instances is a difficult, expensive and time consuming task which, in general, requires human intervention. Moreover, unlabeled data are easier to be collected, however the usage of unlabeled data may be more restricted. According to an evaluation carried out in 2010, only in 2009 the amount of stored data reached about 800000 petabytes [8]. The authors estimate that for 2020 the amount of data created and replicated worldwide is expected to reach 35 million petabytes. Consequently, both the organization and classification of such large amount of data without relying in automatic techniques will become impossible.

The semi-supervised learning (SSL) is a machine learning approach that seeks to learn from labeled and unlabeled data. As the SSL requires less human effort and produces results with high accuracy, it has become an area of great interest and study [6], [28]. These authors emphasize that the most active research field in SSL has been the graph-based methods. These methods require a dataset whose instances are represented by the vertices of a graph. The labeled vertices are used to propagate information to the unlabeled vertices. Thus, these methods generally use a transductive approach in which, unlike the inductive approach, all examples (labeled and unlabeled) are presented simultaneously to the algorithm during the learning phase. In this phase the algorithm "learns" to classify the unlabeled examples presented to it. It is not induced a model for classifying new data as occurs in the inductive approach.

This growing interest in the use of graphs can be justified by the benefits of this representation, since methods based on graphs may: (i) capture the topological structure of the data, (ii) allow a hierarchical representation, i.e., a graph can be partitioned into subgraphs, which in turn can be divided in other subgraphs, and so on, (iii) allow the detection of arbitrary groups or classes [20], (iv) allow the combination of local and global statistics structures [10]. More specifically, data represented in graphs allow the use of collective inference (vertices can affect each other), propagation of labels (autocorrelation among neighbors) and use of neighborhood characteristics of a vertex [13]. Additionally, it allows the use of any formalism of decades of research in graph theory, and currently in complex networks [17].

The main graph-based SSL techniques include the following methods: min-cut [3], Markov random walk [21], propagation of labels [26], Gaussian fields and harmonic functions [27], local and global consistency [25], manifold regularization or Laplacian support vector machine [1], alternating graph transduction [24] and those based on dynamic models [18]. It should be noted, however, that most of the methods described in the literature focuses on developing algorithms for inference of labels without worrying about the construction of the graph and its effect on the base algorithm performance. This question is of great importance and has been raised and studied by some authors such as Whang and Zhang (2008) [23], Maier and Luxburg (2009) [16], Jebara et al. (2009) [11], Talukdar (2009) [22].

On this basis, this paper proposes a novel method for SSL graph construction. This novel technique explores the mutual k-nearest neighbor method (mutual kNN) and the existing labeled data for the graph construction. The mutual kNN favors the construction of a neighborhood graph on the sample such that "most significant clusters" are "identified" [14]. The development of techniques for graph

978-1-4673-4794-5/12/$31.00

construction using prior-information, such as labeled data, is an underexplored topic in the literature [22]. The results show that the proposed technique has better performance in some scenarios compared to traditional k-NN networks and enables graph construction with low average degree. The quantity of edges in a network is related with the processing and physical links cost, so, in some cases it may be interesting to use networks with a smaller number of edges.

II. Definitions and Related Work

Given a set of l labeled instances $L = \{(x_1, y_1), \ldots, (x_l, y_l)\}$ and a set of u unlabeled instances $U = \{x_{l+1}, \ldots, x_{l+u}\}$. The goal of SSL is to infer labels for the set of unlabeled instances U. In most cases, the data instances are assumed to be independent and identically distributed (i.i.d.). At this point, for applying any label propagation technique the common practice is, in a first step, to create a graph from the data instances, and then to apply one of the graph-based SSL methods on the constructed graph.

Given the input data X, a graph construction technique produces a graph $G = (V, E, W)$ consisting of $n = l + u$ vertices V where each vertex V_i is associated with the instance x_i. The set of edges between vertices is represented by E. W is a weighted symmetric adjacency matrix, where each scalar W_{ij} represents the weight of the edge between vertices V_i and V_j.

First, it is necessary to choose a similarity or kernel function for estimating "affinity" between pairs of instances or the weights of the edges between pairs of vertices. Second, it is necessary to choose an algorithm for finding a sparse weighted subgraph. The final step requires the selection of a graph based SSL algorithm, that will diffuse the known labels of the graph to the unlabeled vertices.

The most common algorithm for generating a sparse subgraph is the k-nearest neighbors algorithm (kNN). Each vertex considers its k neighbors using a similarity function and instantiates k undirected edges between itself and these neighbors. There are also the mutual kNN networks in which there is a connection between two vertices only if the rule of the neighborhood is fulfilled by both vertices, i.e., there is an edge between V_i and V_j if and only if $V_i \in k_{neighborhood}(V_j) \wedge V_j \in k_{neighborhood}(V_i)$, where $k_{neighborhood}(V)$ is the set formed by the k nearest neighbors of V. The mutual kNN networks are considered more restrictive and it is traditionally used in unsupervised learning [4], [9]. Another approach is the ϵ-neighborhood graph which includes an undirected edge between two vertices only if the distance between them is smaller than ϵ, where $\epsilon > 0$ is a predefined constant.

For SSL purposes, kNN methods have better properties compared to ϵ-neighborhood based graphs. For example, kNN methods are adaptive to scale and density while an inaccurate choice of ϵ may result in disconnected or densely connected graphs. An empirical verification of this phenomenon on a synthetic dataset is shown in Jebara et al. (2009) [11]. Maier et al. (2008) [15] show that applying the same clustering algorithm on two different graphs constructed using kNN and ϵ-neighborhood may systematically lead to different clustering criteria.

New methods for graph generation were proposed, like the b-matching [11]. The authors propose this technique as an alternative to the kNN graph. Unlike kNN which greedily connects the k nearest neighbors to each vertex and may return graphs where some vertices have many more than k neighbors, b-matching ensures the graph is regular (every vertex with b neighbors). b-matching optimizes the following optimization problem:

$$min_{P \in B} \sum_{i,j} P_{ij} D_{ij}$$

$$s.t. \sum_{j} P_{ij} = b, P_{ii} = 0, P_{ij} = P_{ji}, \forall\ i, j \in 1, \ldots, n$$

where P_{ij} is the adjacency matrix, such that $P_{ij} = 1$ implies an edge between instances x_i and x_j and $P_{ij} = 0$ implies absence of an edge between these instances. D_{ij} is the symmetric distance matrix.

Daitch et al. (2009) [7] trying to find the best graph that fits a set of points, where best is measured as some fitness criteria, propose two types of graphs: hard and α-soft graphs. Hard graphs are graphs where each vertex is strictly required to have (weighted) degree of at least 1. This graph minimizes the following optimization problem:

$$min_w f(W) = \sum_i \|d_i x_i - \sum_j W_{ij} x_j\|^2$$

$$s.t.\ d_i \geq 1, i = 1, \ldots, n$$

where W is the symmetric edge weight matrix and $d_i = \sum_j W_{ij}$ is the weighted degree of a vertex i. It notes that $f(W)$ is similar to the Locally Linear Embedding (LLE) [19] when $d_i = 1, \forall\ i, j$. Sometimes it may be necessary to relax the degree constraint for some vertices, this way, α-soft graphs can be estimated with a new constraint:

$$\sum_i \left(max\left(0, 1 - d_i\right)\right)^2 \leq \alpha n$$

where α is a hyper-parameter. The authors present a variety of experimental results, which demonstrate properties of the constructed hard and 0.1-soft graphs (α = 0.1) as well as their effectiveness in classification, regressions and clustering tasks.

Wang and Zhang (2008) [23] propose the Linear Neighborhood Propagation (LNP) algorithm, which construct a graph by minimizing the equation:

$$min_W \sum_i \|x_i - \sum_j W_{i,j} x_j\|^2$$

$$s.t.\ d_i = \sum_j W_{i,j} = 1, i = 1, \ldots, n$$

Comparing with hard-graph, it is noted that hard graph optimization enforces the constraint $d_i \geq 1$ while the LNP enforces $d_i = 1$. The authors note that the storage requirement of LNP is smaller than traditional graph-based methods. The edge weights of the graph constructed by LNP can be solved in closed form.

Cancho and Sole (2003) [5] use an evolutive algorithm involving minimization of the number of edges and the average shortest path and find four main types of networks: exponential, scale-free, stars and highly dense. To this end, it is used an energy function defined as:

$$E(\lambda) = \lambda d + (1 - \lambda)\rho$$

where $0 \leq \lambda, d, \rho \leq 1$. λ is an hyper-parameter that controls the linear combination between d and ρ, ρ is the number of normalized edges:

$$\rho = \frac{1}{\binom{n}{2}} \sum_{i<j} a_{ij}$$

d is the vertex-vertex normalized distance defined as $d = D/D^{linear}$, where D is the average distance between the vertices:

$$D = \frac{1}{\binom{n}{2}} \sum_{i<j} D_{ij}$$

and $D^{linear} = (n+1)/3$ is the maximum value that can be found in a connected network, assuming a linear graph.

Another approach for graph construction in the supervised context were proposed by Lopes et al. (2009) [12] and Bertini et al. (2011) [2] referred to as k-associated optimal graph (KAOG). Seeking for maximizing a measure of purity of the components given by:

$$P_i = \frac{\langle G_c \rangle}{2k}$$

where k is a parameter to control the number of neighbors used for the graph construction and $\langle G_c \rangle$ correspond to the average degree of component C. The idea is to vary k while maintaining the best components found. This process will result in a network called the optimum network, formed by components with different values of k.

Observing the area literature, Daitch et al. (2009) [7] analyze properties of the induced hard and α-soft graphs, Jebara et al. (2009) [11] emphasize the need for regular graphs in graph-based SSL. Cancho and Sole (2003) [5] explore the construction of networks by optimization. But, as noted by Talukdar (2009) [22] none of the graph construction methods exploit available labeled data during construction of the graph. Labeled data may be seen as a type of prior information which could be useful for improving the graph construction for the current learning task.

It is emphasized that the key to SSL problems is the cluster assumption [6]: (i) nearby points are likely to have the same label (local assumption); (2) points on the same structure, such as a cluster or a submanifold, are likely to have the same label (global assumption). We note that the mutual kNN can better fulfill with these assumption in graph construction and it has not been explored in graph construction for SSL. Hence, we proposed a method that aggregates mutual kNN and labeled data for constructing the graph. This technique is shown in more detail in the next section.

III. Informativity-based graph construction

Initially it is necessary to generate a similarity matrix, using some distance measure, we employ the Euclidean distance. On this matrix we can find the k nearest neighbors of the elements. The graph G is constructed following these steps:

Step 1 $\forall\ x_i \in X$ find its k nearest neighbors and store only its k mutual neighbors. This information is stored in a list of mutual neighbors of x_i, $mutual\_kNN(x_i)$.

Step 2 $\forall\ x_i \in X$ calculate the minimum distance to a labeled point ($min\ d(x_i, k^{th}\ labeled(x))$) and store this information in a vector.

Step 3 Create a connection between x_i and x_j that minimizes the sum of the distances from x_i to its mutual kNN ($d(x_i, mutual\_kNN(x_i))$) and from this $mutual\_kNN(x_i)$ to a labeled vertex ($d(j^{th}\ mutual\_kNN(x_i), k^{th}\ labeled(x))$).

$$min \sum_j d(x_i, mutual\_kNN(x_i)) + d(j^{th}\ mutual\_kNN(x_i), k^{th}\ labeled(x)).$$

This way, the method prioritizes connections between vertices that are mutual neighbors and those closer to labeled instances. We call this technique *informativity-based graph construction*, since it considers information conveyed by labeled vertices and the vertices neighborhood to make the connections.

We emphasize that while kNN graph construction establishes an edge between all k neighbors of a vertex, creating a dense network as the value of k increases, *informativity-based graph construction* can use just one mutual k neighbor to establish connections. In the experiments, good results were obtained using only one neighbor. In this way, the proposed technique optimizes the number of edges, generating less dense networks.

IV. Experiments

The experiments where carried out on artificial and real datasets. The algorithm used for the label inference task was Local and Global Consistency [25]. This algorithm utilizes the graph and W to diffuse the known labels Y_l to all the unlabeled vertices.

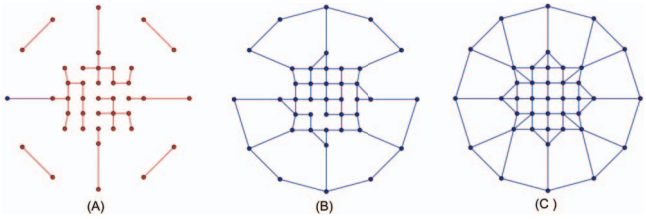

Figure 1. Classification results using kNN graph construction (A) $k = 1$, (B) $k = 2$, (C) $k = 3$.

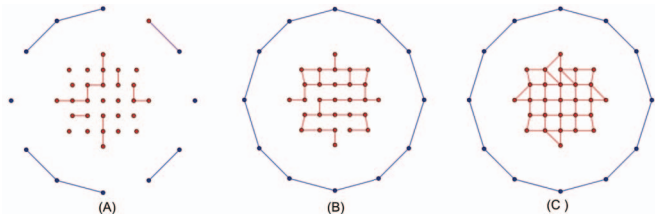

Figure 2. Classification results using *informativity-based graph construction* (A) $k = 2$, (B) $k = 3$, (C) $k = 5$.

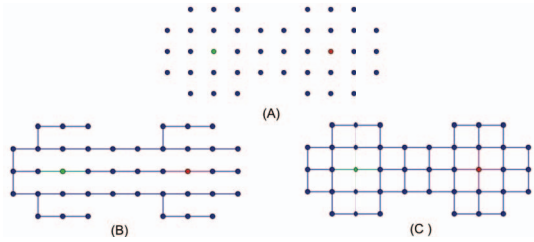

Figure 3. Classification results using kNN graph construction (A) toy dataset, (B) $k = 1$, (C) $k = 2$.

Figures 1 and 2 illustrate points circumscribed by a circle, such that, the internal points have higher density than the external ones. Labeling just one internal and one external point, it is not possible, using kNN graph construction, accurately classify the remaining points, as shown in Figure 1. But, using the proposed technique there is at least one value of k that allows accurate classification of all points, as shown in Figure 2.

Assuming a situation as illustrated in Figure 3 (A), in which the central vertices are labeled. Using kNN graph construction all vertices are connected and consequently classified as the same label, see Figure 3 (B) and (C). Using *informativity-based graph construction*, as the value of k increases, two graphs are formed and this enables the classification of each graph with the label value corresponding to the respective central vertex. Moreover, as k value increases, the network tends to form a hub, in which the labeled vertex is the hub, according to the process illustrated in Figure 4.

This is most clear in Figures 5 and 6. Figure 5 shows the *informativity-based graph construction* applied to a dataset with two Gaussians, in wich one vertex of each Gaussian was initially labeled. Figure 6 shows the *informativity-based graph construction* applied in the Breast Cancer dataset from UCI Machine Learning Repository, in which one vertex of each class was initially labeled. The proposed technique forms two graphs with hubs occurrence, since all vertices seek to connect to the closest labeled vertex.

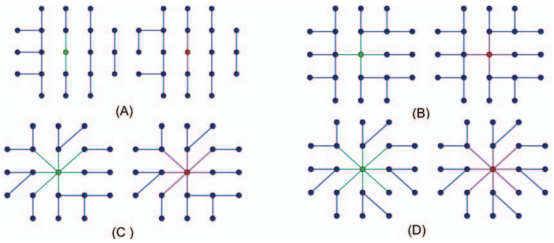

Figure 4. Classification results using *informativity-based graph construction* (A) $k = 3$, (B) $k = 5$, (C) $k = 7$, (D) $k = 10$.

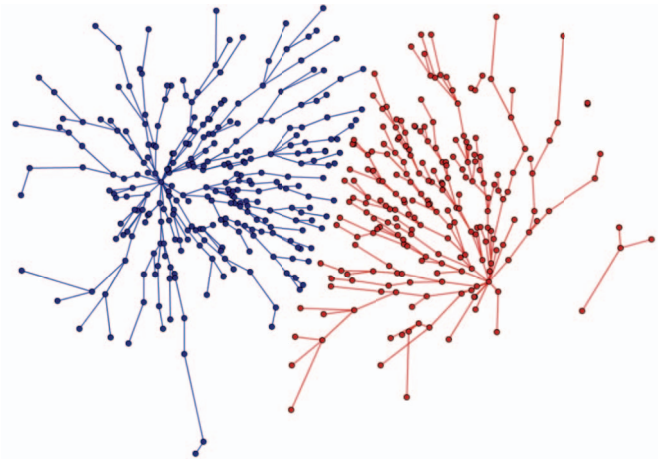

Figure 5. *Informativity-based graph construction* applied to a Gaussian dataset with 500 data points.

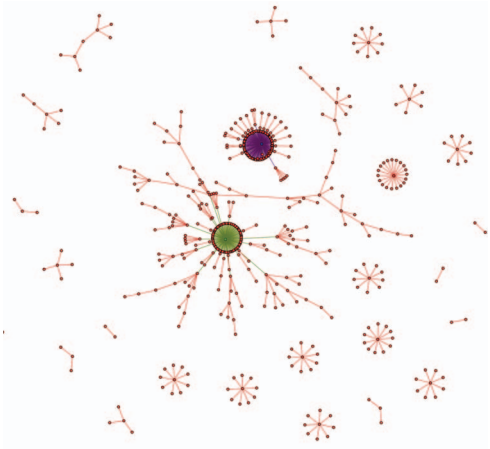

Figure 6. *Informativity-based graph construction* applied to Breast Cancer dataset from UCI.

Figures 7, 8 and 9 show classification accuracy using kNN and *Informativity-based graph construction* applied to

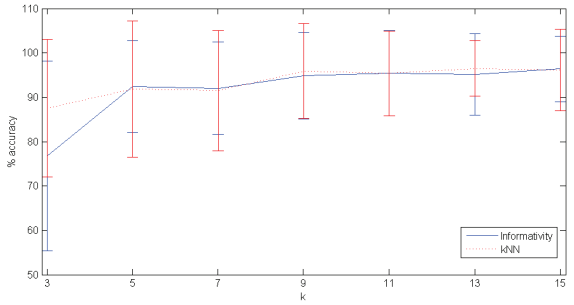

Figure 7. Classification accuracy using kNN and *Informativity-based graph construction* applied to a Gaussian dataset with 500 data points without overlapping classes.

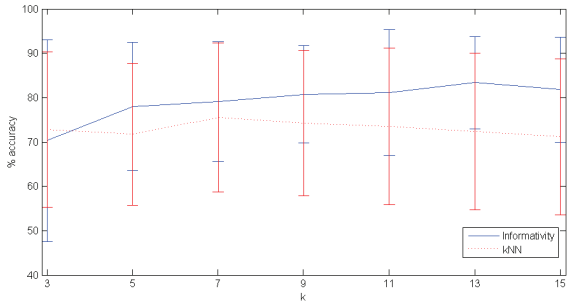

Figure 8. Classification accuracy using kNN and *Informativity-based graph construction* applied to a Gaussian dataset with 500 data points and little overlapping classes.

Gaussian datasets with different levels of classes overlapping (e.g., Figure 5). Just one point was initially labeled in each Gaussian, the labeled vertex was randomly selected and the results are the average of 50 runs. In Figure 7 where do not have overlapping classes (the center of the Gaussians are separated by 5 units), the results obtained by both techniques for graph construction are similar and the accuracy is almost 100%. In Figure 8 where exist some overlapping (the center of the Gaussians are separated by 3 units), the *Informativity-based graph construction* outperforms kNN, as well as in Figure 9, where the overlapping is high (the center of the Gaussians are separated just by 1 unit).

It is observed that larger k values favor the technique, since it allows to achieve more mutual neighbors and thus to produce networks with hub, this favors the diffusion of the labels. Besides, while k value is increased, kNN graph construction forms networks with average degree equals to k, while *Informativity-based graph construction* keeps the networks with average degree equals to 1. That is, among the k mutual neighbors found, each vertex connects to only one vertex: the point that minimize the distance to a labeled vertex and is more similar to x_i.

We carried out some experiments with well known real

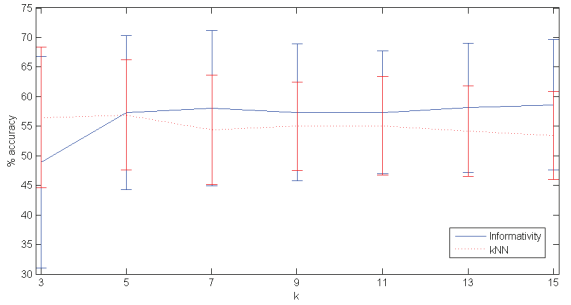

Figure 9. Classification accuracy using kNN and *Informativity-based graph construction* applied to a Gaussian dataset with 500 data points and overlapping classes.

datasets from UCI Machine Learning Repository. For the experiments just one labeled vertex from each class was used. We observe that selecting more than one vertex per class the variance should be reduced. This labeled vertex was randomly selected. The results are the average of 50 runs. In these domains the proposed algorithm outperforms kNN. We emphasize the algorithm achieves better results in data with cluster formations.

Table I
CLASSIFICATION ACCURACY ON REAL DATASETS

Domain	k**nn**	**Informativity**	$\langle G \rangle$ k**nn**	$\langle G \rangle$ **Informativity**
Iris	74 ($\pm$ 12)	**87 ($\pm$ 8)**	5	1
Glass	37 ($\pm$ 10)	**40 ($\pm$ 8)**	9	1
Zoo	64 ($\pm$ 13)	**71 ($\pm$ 15)**	13	1
Ecoli	54 ($\pm$ 15)	**65 ($\pm$ 12)**	11	1
Breast Cancer	82 ($\pm$ 13)	**85 ($\pm$ 9)**	11	1

V. CONCLUSION

In the last years, many graph-based semi-supervised learning algorithms have been proposed, but studies on the influence of graph construction step on such algorithms and new techniques for graph construction have received limited attention. Usually, neighborhood graphs are used for modeling local relationships between data instances. This article proposed a novel technique referred to as *Informativity-based graph construction* that employ information conveyed by mutual k neares neighbors and labeled vertices. Preliminary results suggest that this technique presents advantages when compared with kNN network construction, both in the classification accuracy and on the average degree of the network.

It was observed in the literature, that none of the graph construction methods exploit available labeled data during construction of the graph. So, the proposed technique presents this differential. Besides, mutual kNN allows the identification of "most significant clusters", i.e., a subgraph formed by instances from the same cluster is connected, while subgraphs from different clusters are not connected to each other. We concluded that the technique is promising and

the use of labeled data can be useful in the graph construction for the classification task. As future work, we would like to apply the proposed method for network construction in more scenarios with different SSL algorithms.

ACKNOWLEDGMENT

The authors would like to thank the São Paulo research foundation FAPESP - process 2011/21880-3.

REFERENCES

[1] Belkin, M.; Niyogi, P.; Sindhwani, V. (2005) *On manifold regularization*. International Workshop on Artificial Intelligence and Statistics.

[2] Bertini, J. R.; Zhao, L.; Motta, R.; Lopes, A. A. (2011) *A nonparametric classification method based on K-associated graphs*. Information Sciences, v. 181, p. 1-26.

[3] Blum, A.; Chawla, S. (2001) *Learning from labeled and unlabeled data using graph mincuts*. In Proceedings of the Eighteenth International Conference on Machine Learning, p. 19-26. San Francisco: Morgan Kaufmann.

[4] Brito, M. R.; Chavez, E. L.; Quiroz, A. J.; Yukich, J. E. (1997) *Connectivity of the mutual k nearest neighbor graph in clustering and outlier detection*. Statistics and Probability Letters, v. 35, n.1, p. 33-42.

[5] Cancho, F.; Sole, R. V. (2003) *Optimization in complex networks*. Statistical Mechanics of Complex Networks. Lecture Notes in Physics, v. 625, p. 114-125.

[6] Chapelle, O.; Schlkopf, B.; Zien, A. editors (2006) *Semi-Supervised Learning*. MIT Press, Cambridge, MA.

[7] Daitch, S. I.; Kelner, J. A.; Spielman, D. A. (2009) *Fitting a graph to vector data*. In Proceedings of the 26th Annual International Conference on Machine Learning. ACM New York, NY, USA.

[8] Gantz, J. F.; Reinsel, D. (2010) *The Digital Universe Decade Are You Ready?*. International Data Corporation. IDC White Paper sponsored by EMC.

[9] Gowda, K.; Krishna, G. (1978) *Agglomerative clustering using the concept of mutual nearest neighborhood*. Pattern Recognition, v. 10, n. 2, p. 105-112.

[10] Hein, M.; Audibert, J. Y.; Luxburg, U. V. (2007) *Graph laplacians and their convergence on random neighborhood graphs*. Journal of Machine Learning Research, v. 8, p. 1325-1368.

[11] Jebara, T.; Wang, J.; Chang, S.F. (2009) *Graph construction and b-matching for semi-supervised learning*. In ICML09: Proceedings of the 26th Annual International Conference on Machine Learning, New York, NY, USA: ACM, p. 441-448.

[12] Lopes, A. A.; Bertini, Jr. J. R.; Motta, R.; Zhao, L. (2009) *Classification Based on the Optimal K-Associated Network*. Proceedings of The First International Conference on Complex Sciences: Theory and Applications, p. 1-11.

[13] Macskassy, S.; Provost, F. (2007) *Classification in networked data: A toolkit and a univariate case study*. Journal of Machine Learning Research, v. 8, p. 935-983.

[14] Maier, M.; Hein, M.; Luxburg, U. (2007) *Cluster Identification in Nearest-Neighbor Graphs*. ALT07 Proceedings of the 18th international conference on Algorithmic Learning Theory, p. 196-210.

[15] Maier, M.; Luxburg, U.; Hein, M. (2008) *Influence of graph construction on graph-based quality measures - technical supplement*. http://www.kyb.mpg.de/bs/people/mmaier/nips08supplement.html.

[16] Maier, M.; Luxburg, U. (2009) *Influence of graph construction on graph-based clustering measures*. The Neural Information Processing Systems, v. 22, p. 1025-1032.

[17] Newman, M. (2003) *The structure and function of complex networks*. SIAM Review, v. 45, p. 167-256.

[18] Quiles, M. G.; Zhao, L.; Breve, F. A.; Rocha, A. (2010) *Label propagation through neuronal synchrony*. In: The IEEE 2010 International Joint Conference on Neural Networks, IEEE Computer Press, p. 2517-2524.

[19] Roweis, S.; Saul, L. (2000) *Nonlinear Dimensionality Reduction by Locally Linear Embedding*. Science, v. 290, p. 2323-2326.

[20] Schaeffer, S. (2007) *Graph clustering*. Computer Science Review, v. 1, n. 1, p. 27-63.

[21] Szummer, M. e Jaakkola, T. (2002) *Partially labeled classification with markov random walks*. In Advances in Neural Information Processing Systems, v. 14.

[22] Talukdar, P. P. (2009) *Topics in Graph Construction for Semi-Supervised Learning*. Technical Reports. Department of Computer and Information Science University of Pennsylvania.

[23] Wang, F.; Zhang, C. (2008) *Label propagation through linear neighborhoods*. IEEE Transactions on Knowledge and Data Enginineering, v. 20, p. 55-67.

[24] Wang, J.; Jebara, T.; Chang, S. F. (2008) *Graph transduction via alternating minimization*. International Conference on Machine Learning, p. 1144-1151.

[25] Zhou, D.; Bousquet, O.; Lal, T. N.; Weston, J.; Schlkopf, B. (2004) *Learning with local and global consistency*. In Advances in Neural Information Processing Systems, v. 16, p. 321-328: MIT Press.

[26] Zhu, X.; Ghahramani, Z. (2002) *Learning from Labeled and Unlabeled Data with Label Propagation*. Technical Report CMU-CALD-02-107, Carnegie Mellon University, Pittsburgh.

[27] Zhu, X.; Ghahramani, Z.; Lafferty, J. (2003) *Semi-supervised learning using Gaussian fields and harmonic functions*. In Proceedings of the Twenti-eth International Conference on Machine Learning, p. 912-919.

[28] Zhu, X. (2005) *Semi-supervised learning literature survey*. Technical report 1530 - Computer Sciences, University of Wisconsin-Madison.