Analysis of Graph Construction Methods in Supervised Data Classification

Murillo G. Carneiro
Faculdade de Computação
Universidade Federal de Uberlândia
Uberlândia, Brazil
mgcarneiro@ufu.br

Liang Zhao
Departamento de Computação e Matemática
Universidade de São Paulo
Ribeirão Preto, Brazil
zhao@usp.br

***Abstract*—Graph-based methods have attracted a lot of attention in recent years, especially due to its inherent ability to capture properties of the networked data (e.g., structural and dynamical). Clustering, semi-supervised label propagation and, more recently, data classification are examples of tasks in which graph-based learning methods have obtained relevant results. In any of these tasks, the common approach is (i) to transform the feature vector data in a graph and then (ii) exploit some property uncovered by the network structure. However, most works have focused on the development of models to exploit the graph, while the graph construction step has been little explored. In this article, we conduct a preliminary study to evaluate supervised graph construction methods based on k-nearest neighbors (kNN) and ϵ-radius neighborhood (ϵN) criteria by employing a recently proposed classification technique based on the importance concept of complex networks. Experiments were conducted on artificial and real-world data sets, including the problem of invariant pattern recognition in images. The results show that the graph construction methods under study are able to deal with different configuration of problems (e.g., domain, features, etc). They also suggest that the combination between selective kNN and ϵN is more suitable in data sets with low level of mixture among the classes, while kNN seems slightly better in problems with higher noise levels.**

***Index Terms*—Graph construction, Network formation, Data Classification, Importance-based classification, PageRank, Complex Networks, Supervised Learning**

I. Introduction

Data classification is one of the most important tasks in machine learning. Medical disease diagnosis, income tax fraud, spam detection and digit recognition are common examples of problems addressed as a classification task. Formally, a data classification problem is composed by a set $\mathcal{X}$ of labeled objects, in which each object $i \in \mathcal{X}$ is denoted by the following tuple (x_i, y_i), with $x_i = \{f_1, \ldots, f_d\}$ denoting the d-dimensional input features of object i and $y_i \in \{1, \ldots, C\}$ the class label associated with it according to an unknown function (oracle, expert, etc) $f(x) \rightarrow y$. Data classification has commonly two phases: training and test. In the training phase, it aims to build a classifier, i.e., learn a known function f', which maps the input features to the class $f'(x) \rightarrow y$, such that $f' \approx f$. In the test phase, a new object $j \in \mathcal{U}$ which the class is unknown, i.e., denoted by the tuple $(x_j, ?)$, is labeled by f'. Notice that an unbiased learning implies that $\mathcal{U} \cap \mathcal{X} = \emptyset$.

Despite that the most known examples of f' functions include some kind of learning based only on the physical features of the data (e.g., distribution or distance), such as neural networks, decision trees, instance-based methods, support vector machines, etc, there are also recent classification techniques which also considers topological patterns of the data by using graphs [1]. Formally, the set of $\mathcal{X}$ labeled objects is transformed into a graph $\mathcal{G} = \{V, E\}$ according to some graph construction function g, i.e., $g(\mathcal{X}) \rightarrow \mathcal{G}$; each vertex $v_i \in V$ denotes a labeled instance $i \in \mathcal{X}$ and each edge $e_{i,u} \in E$ a link between vertices $v_i, v_u \in \mathcal{X}$. Graph-based learning is usually associated to a function f' which exploits some information (e.g., spatial, structural or dynamical) of the networked data. Thus, $\mathcal{G}$ plays a key role in the predictive results as the pattern formation of the classes are directly extracted from it.

In order to extract relevant properties of the networked data, complex networks concepts and measures have been largely employed in machine learning to provide structural analysis of the data, especially in tasks such as clustering, dimensionality reduction and semi-supervised label propagation [2]–[7] . On the other hand, the usage of complex networks in graph-based supervised classification has been a recently explored topic. In the related literature, [1] proposed a hybrid classification framework to consider the semantic relations among the data instances by combining the associations produced by traditional (e.g., neural networks) and network-based techniques, with this latter using a set of complex network measures to estimate the membership of a test item according to the data pattern formation. In [8] is proposed a technique that performs classification by calculating the limiting probabilities of the random walk theory over an adjacency matrix which carries both physical and structural information about the data. Another related work is [9] which proposed a new data classification concept based on the importance concept of complex networks, in which instead of data space division as having been done in traditional techniques or pattern conformation as having been done in high-level classification techniques, the classification considers the individual importance of each data item in order to classify an unlabeled item into that class where it has the highest importance.

Although the above mentioned works have considerable

contribution in terms of graph exploitation, they do not focus on graph construction methods. Indeed, there are very few reported graph construction methods proposed to directly handle vector-based data in the supervised learning literature [1], [10], while a reasonable amount of studies has proposed graph methods for unsupervised, semi-supervised and dimensionality reduction tasks (e.g., [11]–[19]). Thus, the supervised works usually employ simple graph construction methods like the k-nearest neighbors (kNN) [20] or one of its variations, such as the combination between kNN and ϵ-neighborhood (ϵN) criteria [1] or the k-associated optimal graph [10], [21], [22]. Recent works have also introduced the usage of nature-inspired algorithms in order to construct optimized graphs for supervised data classification [23].

Due to the lack of comparative analysis among supervised graph construction methods, this paper presents a preliminary study in this sense. By adopting the recently proposed importance-based classification technique [9], we evaluate the graphs provided by four supervised graph construction methods based on k-nearest neighbors and ϵ-radius neighborhood criteria. Such methods are evaluated over artificial and real-world data sets, including the problem of invariant pattern recognition in images.

The remainder of the paper is organized as follows. Sect. II describes the importance-based classification as well as the graph construction methods under study. Sect. III presents the experimental results obtained by our models over artificial and real-world data sets. Finally, Sect. IV concludes the paper.

II. Models Description

In this section we present an overview about the methods investigated. Sub-sect. II-A introduces the importance-based classification and Sub-sect. II-B describe in details the graph construction methods under study.

A. Importance-based Classification

The importance-based classification is a recently proposed technique based on complex network measures which considers the individual importance of each data item in order to classify an unlabeled item into that class where it has the highest importance [9]. In the technique, the concept of importance is derived from PageRank, the ranking measure operating behind the universal search engine of Google. For short, hereafter, we will refer to this technique as PGR as it is derived from PageRank.

PGR can be divided in two phases:

- **Training phase.** In the training phase, the technique builds up a network $\mathcal{G}$ by employing a graph construction method g over a set $\mathcal{X}$ of labeled items and then exploits $\mathcal{G}$ by calculating the efficiency patterns $\mathcal{E}$ of its components and also the individual importance I of each vertex (given by the PageRank formulation).
- **Test phase.** In the testing phase, a test item $j \in \mathcal{U}$ is presented to PGR, which selects vertices to temporarily connect to j by considering a measure named spatio-structural differential efficiency. After that, the importance score of j to each class is calculated. At the end, j is classified into that class where it has more importance.

Formally, the importance I of a test instance j with respect to a given class $\mathcal{L}$ is given by:

$$I_j^{(\mathcal{L})} = \sum_{v_i \in \Lambda_j^{\mathcal{L}}} I_i \ , \tag{1}$$

where $v_i \in X_{train}$ denotes a labeled node, $\Lambda_j^{\mathcal{L}}$ is the set of nodes pertaining to the class $\mathcal{L}$ where j is temporarily connected, and I_i means the importance of node v_i (calculated from PageRank formulation [24]). In addition, the set of temporary links to j is obtained from the following formulation:

$$\Lambda_j^{\mathcal{L}} \cup \{v_i \mid \mathcal{F}_{j,i} \geq 0 \ \textit{and} \ y_i \in \mathcal{L}\} \tag{2}$$

where $\mathcal{F}_{j,i}$ denotes the spatio-structural differential efficiency, which adds vertex i to $\Lambda_j^{\mathcal{L}}$ if a link between j and i increases the efficiency of its component. $\mathcal{F}_{j,i}$ is given by:

$$\mathcal{F}_{j,i} = \underset{v_i \in \alpha}{\mathcal{E}^{\alpha}} . \gamma - D_{j,i} \ , \tag{3}$$

where $D_{y,j}$ refers to the distance between nodes y and j, $\mathcal{E}^{\alpha}$ is the efficiency of the component α, and γ is a parameter which fits the component efficiency values obtained from the network formation method. The efficiency of a component $\mathcal{E}^{\alpha}$ can be defined as the average value of the local efficiency ξ of the nodes that belong to α, i.e.,

$$\mathcal{E}^{\alpha} = \frac{1}{N^{\alpha}} \sum_{i \in \alpha} \xi_i^{(\alpha)} \ , \quad \xi_i^{(\alpha)} = \frac{1}{N_i} \sum_{i \to j} D_{i,j} \tag{4}$$

where N^{α} denotes the number of nodes in the component α, N_i denotes the number of links from i and $D_{i,j}$ is the proximity measure between nodes i and j. Please refer to [9] for more details about PGR.

B. Supervised Graph Construction Methods

In the machine learning literature, the most common graph construction methods are the k-nearest neighbors and ϵ-radius neighborhood [11]. The former creates a link from vertex v_i to v_u if v_u is one of the k nearest neighbors of v_i; and the latter creates a link between vertex v_i and v_u if the distance (or similarity) between them is less than a pre-defined value ϵ. In case of supervised learning, there are also the class label of the training data which needs to be the same between v_i and v_u, i.e., $y_i = y_u$ [10].

In the following, we define the four graph construction methods analyzed in this preliminary study. Formally, let $\mathbf{S}$ be a distance matrix in which $\mathbf{S}_{iu} = \delta(x_i, x_u)$ and $k\text{NN}(x_i)$ be the set of k nearest neighbors of x_i.

1) kNNG: The adjacency matrix $\mathbf{A}$ of a kNN graph is:

$$\mathbf{A}_{iu} = \begin{cases} 1, & \text{if } x_u \in k\text{NN}(x_i) \text{ and } y_i = y_u \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

2) kNN+ϵNG: The adjacency matrix $\mathbf{A}$ of the combination of a kNN and ϵN graphs is given by:

$$\mathbf{A}_{iu} = \begin{cases} 1, & \text{if } x_u \in k\text{NN}(x_i) \text{ and } y_i = y_u \\ 1, & \text{if } \mathbf{S}_{iu} < \epsilon \text{ and } y_i = y_u \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

Alternatively, it is also possible to remove the comparison between the objects classes by calculating the distances only among objects of the same class, which we called Selective kNN (Sel-kNN). Besides the methods already listed, we also evaluate these below.

3) Sel-kNNG: The adjacency matrix $\mathbf{A}$ of a Selective kNN graph is given by:

$$\mathbf{A}_{iu} = \begin{cases} 1, & \text{if } x_u \in \text{Sel-}k\text{NN}(x_i) \\ 0, & \text{otherwise.} \end{cases} \quad (7)$$

4) Sel-kNN+ϵNG: The adjacency matrix $\mathbf{A}$ of the combination of Selective kNN and ϵN graphs is obtained as follows:

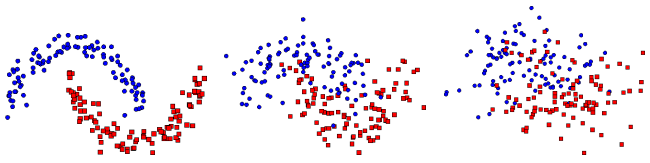
$$\mathbf{A}_{iu} = \begin{cases} 1, & \text{if } x_u \in \text{Sel-}k\text{NN}(x_i) \\ 1, & \text{if } \mathbf{S}_{iu} < \epsilon \text{ and } y_i = y_u \\ 0, & \text{otherwise.} \end{cases} \quad (8)$$

III. Experiments

In this section we present experimental results to evaluate the graph construction methods under study. Sub-sect. III-A and III-B discuss respectively results obtained in artificial and real-world data sets. Sub-sect. III-C describes the application of the methods in the detection of invariant patterns in images.

A. Artificial Data Sets

In the following, we present an analysis of the graph construction methods in three artificial data sets with distinct levels of noise which are shown by Fig. 1.

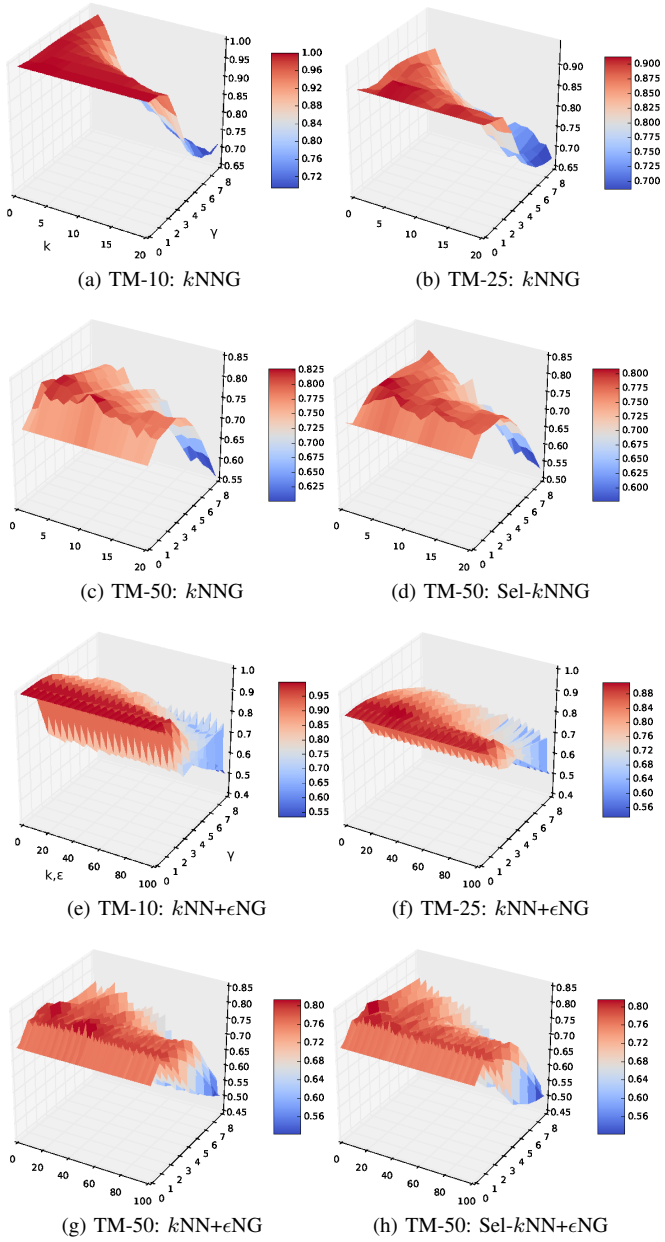


(a) TM-10 (b) TM-25 (c) TM-50

Fig. 1: Artificial data sets generated with distinct level of mixture between the classes. (a) 10%; (b) 25% and (c) 50%.

In the experiments, the predictive performance of each method is averaged over a 10-fold stratified cross-validation. The parameters are defined as follows: the number of neighbors $k \in \{1, 2, \ldots, 20\}$; the radius $\epsilon \in \{0.2\bar{d}, 0.4\bar{d}, \ldots, 1\bar{d}\}$, where $\bar{d}$ denotes the average distance among objects in the training data set; and the $\gamma \in \{0, 2^{-4}, 2^{-3}, \ldots, 2^{3}\}$, which is defined by Eq. (3).

Fig. 2 presents the results of the parameter analysis. Each subfigure presents the averaged accuracy (axis z) obtained by a given method in a given data set; axes x and y denote



(a) TM-10: kNNG (b) TM-25: kNNG
(c) TM-50: kNNG (d) TM-50: Sel-kNNG
(e) TM-10: kNN+ϵNG (f) TM-25: kNN+ϵNG
(g) TM-50: kNN+ϵNG (h) TM-50: Sel-kNN+ϵNG

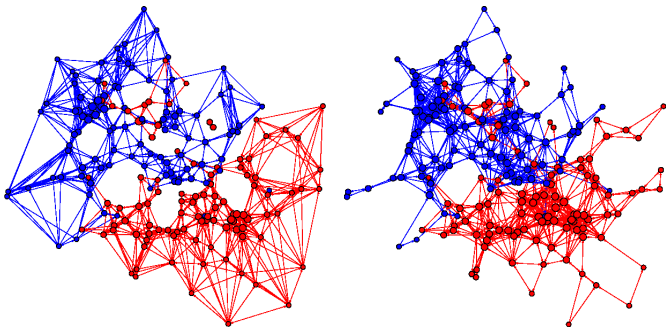
Fig. 2: Analysis of the graph construction methods in function of the parameters.

respectively the parameters of PGR and of the graph construction method under evaluation. As the general performance of kNNG and Sel-kNNG are similar for data sets with TM-10 and TM-25, we show the results of only one of them (the same with kNN+ϵNG and Sel-kNN+ϵNG). One can see in the figure that the γ parameter has an important role as the noise data increase. Table I shows the best results obtained for each method in each data set. By the table, one can see that kNNG seems slightly more robust to noise than Sel-kNNG. This also has some evidence by analyzing the networks provided by each method, such as shown by Fig. 3: while kNNG has some isolated vertices (maybe noise data), Sel-kNN+ϵNG forms

only connected components.

TABLE I: Best averaged accuracy obtained by PGR using each graph construction method.

Data	kNNG	kNN+ϵNG	Sel-kNNG	Sel-kNN+ϵNG
TM-10	100.0	100.0	100.0	100.0
TM-25	92.5	92.5	92.5	92.0
TM-50	84.0	84.0	82.0	83.5



(a) TM-50: kNNG (b) TM-50: Sel-kNN+ϵNG

Fig. 3: Analysis of the graphs generated by distinct graph construction methods in the artificial data sets.

B. Real-world Data Sets

We also conducted simulations on real-world data sets available in the UCI repository [25]. Table II provides details about each data set. As one can see, the selection was made to encompass diversity of domains, features and classes. As a data preparation, each instance attribute vector was normalized by the *l2*-norm and the Euclidean distance was used in all simulations as the distance measurement.

TABLE II: Description of the data sets in terms of the number of data items (*#Obj.*), attributes (*#Attr.*) and classes (*#Classes*).

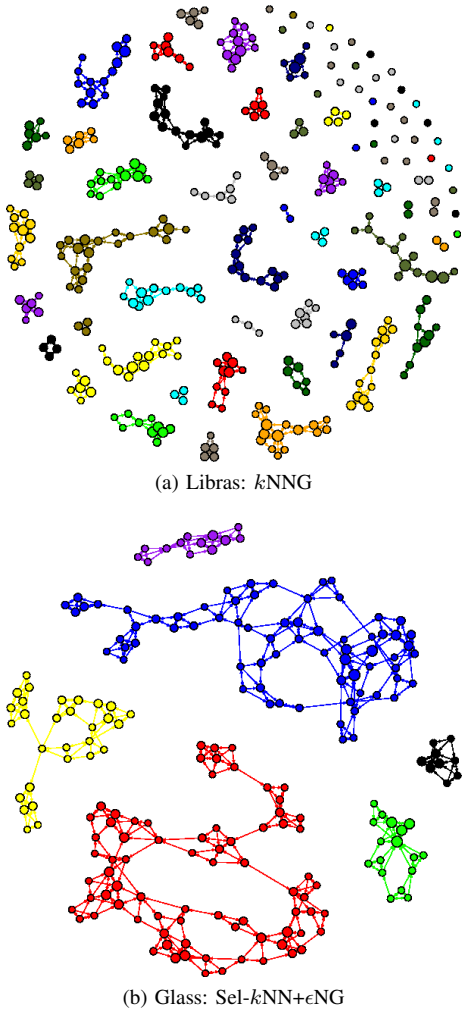
Name	#Obj.	#Attr.	#Classes
Iris	150	4	3
Teaching	151	5	3
Glass	214	9	6
Libras	360	91	15

In our study, the predictive performance of the methods is averaged over a repeated stratified cross-validation that averages three runs of 10-fold stratified cross-validation, taking the folds randomly each time. The parameters of each technique (presented in Sub-sect. III-A) are selected through the grid search method by doing a 3-fold stratified cross-validation on each training partition (nested cross-validation).

Table III shows the results obtained by each method in terms of averaged accuracy. It also includes as baseline the results of the 1-nearest neighbor classifier (1NN). Both kNNG and Sel-kNN+ϵNG performed better in two data sets. In order to better understand such results, we analyze some of the obtained graphs. Figure 4 presents a graph obtained by kNNG in the Libras data set and a graph obtained by Sel-kNN+ϵNG in the Glass data. Despite both graphs have particular structures, again the network provided by kNNG has a set of isolated vertices (maybe noise data). By contrast, the Sel-kNN+ϵNG presents only connected components, which allows the representation of strong and well-defined patterns, although it can also suffer with noise data.

TABLE III: Predictive results obtained by each graph construction method in PGR. 1NN provides baseline results.

Data	kNNG	kNN+ϵNG	Sel-kNNG	Sel-kNN+ϵNG	1NN
Iris	97.3 ± 4.1	97.3 ± 4.1	96.7 ± 4.1	**97.6 ± 3.2**	**97.6 ± 3.6**
Teach.	**60.9 ± 10.**	58.2 ± 12.	60.0 ± 13.	58.0 ± 15.	39.7 ± 11.
Glass	71.4 ± 8.5	**72.1 ± 7.7**	71.2 ± 8.4	**72.1 ± 7.9**	67.9 ± 8.4
Libras	**85.3 ± 5.4**	84.8 ± 5.8	84.7 ± 5.2	84.3 ± 6.1	75.6 ± 5.2



(a) Libras: kNNG

(b) Glass: Sel-kNN+ϵNG

Fig. 4: Analysis of the graphs generated by distinct graph construction methods in the real-world data sets. Each color denotes a class label.

C. Invariant Object Recognition

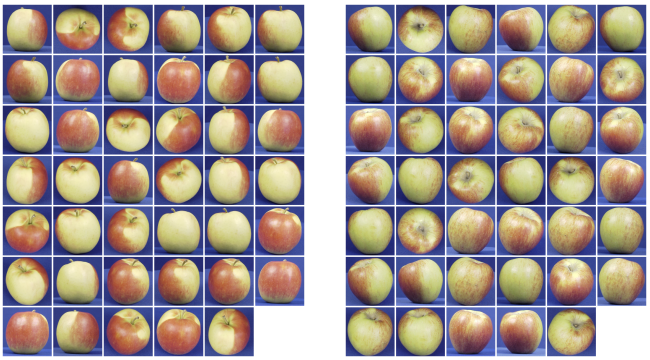
In the following the graph construction methods are evaluated in the invariant object recognition problem. Instead of using the Euclidean distance, here the discrete Bhattacharyya coefficient is employed to compute the similarities between images as it is widely used in image processing tasks.

The data sets are from the ETH-80 collection [26], which comprises a total of 3280 images divided in 8 categories: Apple, Car, Cow, Cup, Dog, Horse, Pear and Tomato, as shown by Fig. 5. Each category contains 10 objects that span large in-class variations while still clearly belonging to the category. For each object, there are 41 images from viewpoints spaced equally over the upper viewing hemisphere (at distances from 22.5^o to 26.0^o). For instance, Fig. 6 shows the 41 images of two apples objects from Apple category. To be specific, the data preprocessing consisted of the following steps: the images were down-sampled from 128×128 (original size) to 32×32 to speed up processing; a total of 512 features were extracted for each image by calculating its histogram; and the similarities among the images' histograms are calculated by the Bhattacharyya coefficient.

Fig. 5: The eighty objects of the ETH-80 collection.

The predictive performance of the methods is averaged over a repeated stratified cross-validation that averages three runs of 10-fold stratified cross-validation, taking the folds randomly each time. About the parameters of the methods (presented in Sub-sect. III-A), they are selected through the grid search method by doing a 3-fold stratified cross-validation on each training partition (nested cross-validation).

Table IV shows the predictive results obtained by PGR for each graph construction method. One can see in the table that the Selective kNN graph has better performance in these data sets than the conventional kNN graph. By analyzing some of the obtained graphs, such as those shown by Fig. 7, we have some evidences that ETH-80 data sets are characterized by strong patterns and low level of noises, which seems to be a favorable scenario to the Selective-kNN method.



(a) Apple category - Object 2 (b) Apple category - Object 7

Fig. 6: Two objects of the Apple category in ETH-80 collection.

TABLE IV: Predictive results obtained by each graph construction method in PGR. 1NN provides baseline results.

Data	kNNG	kNN+ϵNG	Sel-kNNG	Sel-kNN+ ϵNG	1NN
Apple	90.2 ± 4.3	90.2 ± 5.1	**90.4 ± 4.7**	**90.4 ± 4.7**	86.2 ± 4.9
Car	94.9 ± 3.6	95.0 ± 3.6	**95.3 ± 4.1**	**95.3 ± 4.1**	90.1 ± 4.1
Cow	**73.6 ± 6.2**	73.4 ± 6.9	72.3 ± 8.6	72.9 ± 9.0	66.8 ± 5.9
Cup	94.9 ± 3.9	95.4 ± 2.9	**95.6 ± 3.4**	95.5 ± 3.5	89.8 ± 4.1
Dog	91.2 ± 3.6	**91.9 ± 2.7**	91.5 ± 3.2	**91.9 ± 3.1**	84.9 ± 6.3
Horse	90.1 ± 3.5	**91.0 ± 2.8**	90.7 ± 3.7	90.5 ± 3.7	81.2 ± 6.8
Pear	80.2 ± 4.6	80.2 ± 3.6	**81.1 ± 3.4**	80.8 ± 3.3	74.7 ± 6.5
Tomat.	93.3 ± 3.8	93.7 ± 5.0	**94.0 ± 3.4**	**94.0 ± 3.4**	89.9 ± 4.8

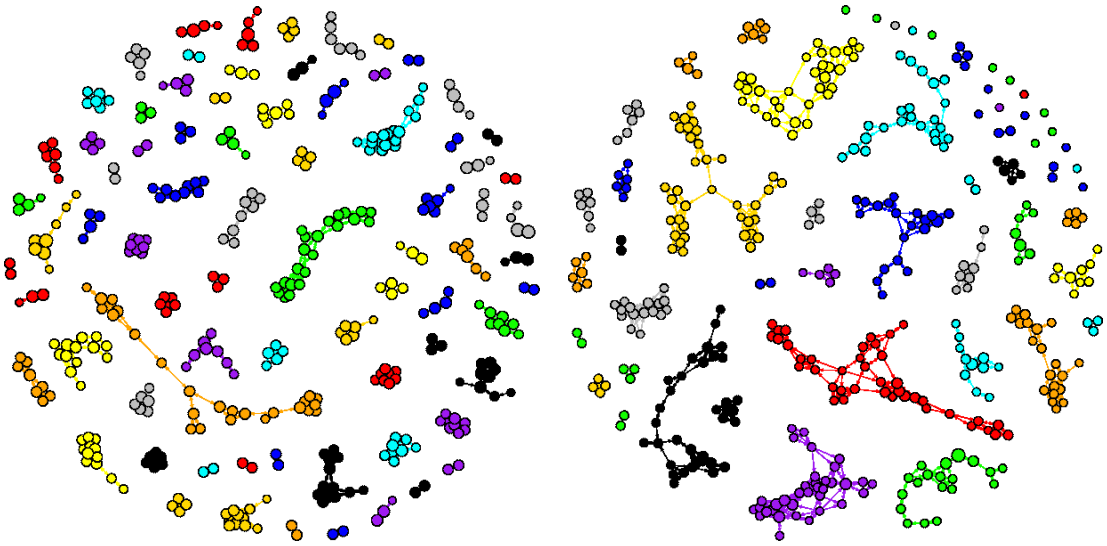
IV. Conclusions

In this paper is presented a preliminary study which addresses the problem of graph construction in supervised learning. In a few words, four graph construction methods based on k-nearest neighbors and ϵ-radius neighborhood criteria were modeled and eqquiped with the recently proposed importance-based classification technique. Experiments were performed considering artificial and real-world data sets. The results obtained show that the inherent structure of the k-nearest neighbors graph makes the method more robust against noises in comparison with other methods evaluated. On the other hand, the combination of the ϵ neighborhood graph with the Selective kNN graph seems interesting to data sets with well-defined and strong patterns.

For sake of space, we reduced the amount of variables in this study to four graph construction methods. However, future works should include the comparison with other graph construction methods in literature as well as the analysis of additional variables, such as the proximity measure, the supervised technique, etc. Future works also include the design of a benchmark data set able to characterize desired properties related to complex networks in graph-based learning.

Acknowledgment

This work was funded by the São Paulo State Research Foundation - FAPESP (grant numbers 2012/07926-3, 2013/07375-0 and 2015/50122-0).



(a) Car: Sel-kNN+ϵNG (b) Cow: kNNG

Fig. 7: Analysis of the graphs generated by distinct graph construction methods in data sets from the ETH-80 collection.

REFERENCES

[1] T. C. Silva and L. Zhao, "Network-based high level data classification," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 6, pp. 954–970, 2012.

[2] M. G. Quiles, L. Zhao, R. L. Alonso, and R. A. F. Romero, "Particle competition for complex network community detection," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 18, no. 3, p. 033107, 2008.

[3] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3–5, pp. 75 – 174, 2010.

[4] M. E. Newman, "Communities, modules and large-scale structure in networks," *Nature Physics*, vol. 8, no. 1, pp. 25–31, 2012.

[5] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 40 –51, 2007.

[6] X. Zhu, "Semi-supervised learning literature survey," Tech. Rep., 2008.

[7] T. C. Silva and L. Zhao, "Network-based stochastic semisupervised learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 3, pp. 451–466, 2012.

[8] T. H. Cupertino, M. G. Carneiro, Q. Zheng, J. Zhang, and L. Zhao, "A scheme for high level data classification using random walk and network measures," *Expert Systems with Applications*, vol. 92, pp. 289–303, 2017.

[9] M. G. Carneiro and L. Zhao, "Organizational data classification based on the importance concept of complex networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 8, pp. 3361–3373, 2018.

[10] J. R. Bertini, L. Zhao, R. Motta, and A. A. Lopes, "A nonparametric classification method based on k-associated graphs," *Information Sciences*, vol. 181, no. 24, pp. 5435–5456, 2011.

[11] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[12] T. Jebara, J. Wang, and S.-F. Chang, "Graph construction and b-matching for semi-supervised learning," in *International Conference on Machine Learning*, 2009, pp. 441–448.

[13] L. Zhang, L. Qiao, and S. Chen, "Graph-optimized locality preserving projections," *Pattern Recognition*, vol. 43, no. 6, pp. 1993–2002, 2010.

[14] C. A. R. de Sousa, S. O. Rezende, and G. E. Batista, "Influence of graph construction on semi-supervised learning," in *Machine Learning and Knowledge Discovery in Databases*. Springer, 2013, pp. 160–175.

[15] T. H. Cupertino, M. G. Carneiro, and L. Zhao, "Dimensionality reduction with the k-associated optimal graph applied to image classification," in *IEEE International Conference on Imaging Systems and Techniques*, 2013, pp. 366–371.

[16] M. G. Carneiro, T. H. Cupertino, and L. Zhao, "K-associated optimal network for graph embedding dimensionality reduction," in *IEEE International Joint Conference on Neural Networks*, 2014, pp. 1660–1666.

[17] L. Berton and A. d. A. Lopes, "Graph construction based on labeled instances for semi-supervised learning," in *International Conference on Pattern Recognition*, 2014, pp. 2477–2482.

[18] B. Araujo and L. Zhao, "Data heterogeneity consideration in semi-supervised learning," *Expert Systems with Applications*, vol. 45, pp. 234–247, 2016.

[19] M. G. Carneiro, T. H. Cupertino, R. Cheng, Y. Jin, and L. Zhao, "Nature-inspired graph optimization for dimensionality reduction," in *IEEE International Conference on Tools with Artificial Intelligence*, 2017, pp. 1113–1119.

[20] T. H. Cupertino, L. Zhao, and M. G. Carneiro, "Network-based supervised data classification by using an heuristic of ease of access," *Neurocomputing*, vol. 149, pp. 86–92, 2015.

[21] M. G. Carneiro and L. Zhao, "High level classification totally based on complex networks," in *IEEE BRICS Congress on Computational Intelligence*, 2013, pp. 507–514.

[22] M. G. Carneiro, J. L. G. Rosa, A. A. Lopes, and L. Zhao, "Network-based data classification: combining k-associated optimal graphs and high-level prediction," *Journal of the Brazilian Computer Society*, vol. 20, no. 1, pp. 1–14, 2014.

[23] M. G. Carneiro, L. Zhao, R. Cheng, and Y. Jin, "Network structural optimization based on swarm intelligence for highlevel classification," in *IEEE International Joint Conference on Neural Networks*, 2016, pp. 3737–3744.

[24] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: bringing order to the web," Tech. Rep., 1999.

[25] D. Dheeru and E. Karra Taniskidou, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[26] B. Leibe and B. Schiele, "Analyzing appearance and contour based methods for object categorization," in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, vol. 2. IEEE, 2003, pp. II–409.