Classification via Regularization on Graphs

Aliaksei Sandryhaila

Department of ECE
Carnegie Mellon University
Pittsburgh, PA, USA
asandryh@andrew.cmu.edu

José M. F. Moura

Department of ECE
Carnegie Mellon University
Pittsburgh, PA, USA
moura@ece.cmu.edu

***Abstract*—We present a novel data classifier that is based on the regularization of graph signals. Our approach is based on the theory of discrete signal processing on graphs where the graph represents similarities between data and we interpret labels for the dataset elements as a *signal* indexed by the nodes of the graph. We postulate that true labels form a low-frequency graph signal and the classifier finds the smoothest graph signal that satisfies constraints given by known data labels. Our experiments demonstrate that our approach achieves high accuracy in multiclass classification and outperforms other classification approaches.**

***Index Terms*—Discrete signal processing on graphs, graph signal, graph shift, total variation on graphs, regularization, classification.**

I. Introduction

Classification and data labeling are important problems in machine learning and data mining, [1]. Classification categorizes elements of a dataset into two or more groups based on specific parameters. For example, documents may be attributed to different classes based on their topic; image databases can be classified based on their content; and customers may be assigned to a group based on their shopping preferences.

It is not feasible or practical to classify large datasets manually. A common semi-supervised learning approach represents a dataset with a graph, where nodes correspond to dataset elements and edges represent similarities. assumes known only a subset of data element labels, and then uses the structure of the graph to predict the missing labels. A central assumption in this approach is that similar dataset elements tend to be in the same class, so knowledge of their similarity should allow inferring unknown labels from known ones. Many labeling and clustering algorithms, e.g., spectral learning or Laplacian eigenmaps, are based on this assumption, [2], [3], [4].

We study classification of large datasets from the perspective of discrete signal processing on graphs (DSP_G)—a novel framework that represents complex datasets as signals indexed by graphs and defines fundamental signal processing concepts for such signals [5], [6], [7]. DSP_G extends to graph signals and graph filters classical signal processing concepts as well as algebraic signal processing, [8], [9], [10], [11]. In this paper, we apply to image and document datasets a multiclass classifier that searches for a smooth graph signal that is conditioned on initially known labels [12]. The classifier finds the graph signal that minimizes the total variation of a graph signal, see [12], subject to the known label constraints. Our experiments demonstrate that the classifier achieves higher classification accuracy than other approaches, such as support vector machines and neural networks, two widely used techniques, as well as a classification method based on Laplacian matrices of similarity graphs for datasets.

II. Discrete Signal Processing on Graphs

We briefly review needed concepts from DSP_G theory from [5], [6], [7], [12].

A. Graph Signals

In DSP_G, a dataset is represented with a graph $G = (\mathcal{V}, \mathbf{A})$, where $\mathcal{V} = \{v_0, \ldots, v_{N-1}\}$ is the set of nodes and $\mathbf{A}$ is the weighted adjacency matrix of the graph. Each data element corresponds to node v_n, and each weight $\mathbf{A}_{n,m}$ of a directed edge from v_m to v_n reflects the degree of relation (e.g., similarity or dependency) of the mth data element to the nth one. The dataset is viewed as a *graph signal* indexed by the graph G and defined as a map from the set $\mathcal{V}$ of nodes to the set of complex numbers $\mathbb{C}$:

$$\begin{aligned} \mathbf{s} \;&:\; \mathcal{V} \to \mathbb{C}, \\ &\quad\; v_n \mapsto s_n. \end{aligned} \tag{1}$$

The mapping (1) can also be written as

$$\mathbf{s} = \begin{pmatrix} s_0 & s_1 & \ldots & s_{N-1} \end{pmatrix}^T \in \mathbb{C}^N,$$

where each element s_n is *indexed* by node v_n of a given representation graph $G = (\mathcal{V}, \mathbf{A})$, as defined by (1).

B. Graph Filters

A *graph filter* $\mathbf{H}(\cdot)$ takes a graph signal $\mathbf{s}$ as input and produces as output the graph signal $\tilde{\mathbf{s}} = \mathbf{H}(\mathbf{s})$. A basic nontrivial graph filter is the *graph shift* defined by the local operation of replacing a signal value s_n at node v_n with a linear combination of elements at its neighbors. The output of the graph shift is given by the product of the input signal with the adjacency matrix of the graph:

$$\tilde{\mathbf{s}} = \mathbf{A}\mathbf{s}. \tag{2}$$

A linear, shift-invariant (i.e., commuting) graph filter is a polynomial in the adjacency matrix $\mathbf{A}$:

$$h(\mathbf{A}) = h_0\,\mathbf{I} + h_1\mathbf{A} + \ldots + h_L\mathbf{A}^L. \tag{3}$$

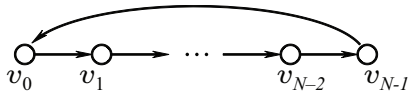

Fig. 1. Traditional graph representation for a finite discrete periodic time series of length N.

Example: Discrete time series. Finite discrete periodic time series are commonly represented with the directed cycle graph shown in Fig. 1 [9], [5]. The edge from the last vertex v_{N-1} to the first v_0 captures the periodicity assumption $s_N = s_0$. The adjacency matrix of this graph is the $N \times N$ circulant matrix

$$\mathbf{A} = \mathbf{C}_N = \begin{pmatrix} & & & 1 \\ 1 & & & \\ & \ddots & & \\ & & 1 & \end{pmatrix}. \tag{4}$$

The time shift is represented by the time delay:

$$\widetilde{s}_n = s_{(n-1) \mod N}.$$

In matrix-vector form, it is written as

$$\tilde{\mathbf{s}} = \mathbf{C}_N \, \mathbf{s}. \tag{5}$$

III. Total Variation on Graphs

In [12], we introduce the concept of signal variation on graphs and define low and high frequencies for graph signals.

The *total variation on a graph* ($\mathrm{TV_G}$) of a signal $\mathbf{s}$ indexed by a graph $G = (\mathcal{V}, \mathbf{A})$ is defined as

$$\mathrm{TV_G}(\mathbf{s}) = \frac{1}{||\mathbf{s}||_2^2} \left|\left| \mathbf{s} - \frac{1}{|\lambda_{max}|} \mathbf{A}\mathbf{s} \right|\right|_2^2. \tag{6}$$

Here, λ_{max} is the largest-magnitude eigenvalue of $\mathbf{A}$ that satisfies the condition $|\lambda_{max}| \geq |\lambda_m|$ for any $0 \leq m \leq M-1$.

The definition (6) extends the concept of total variation from regular lattices that are used for time and space signals, such as the one in Fig. 1. In classical signal processing, the total variation of a discrete signal is defined as the sum of the magnitudes of the differences between two consecutive signal samples [13]:

$$\mathrm{TV}(\mathbf{s}) = \sum_n \left| s_n - s_{n-1} \right|. \tag{7}$$

When $\mathbf{s}$ is a discrete periodic time series of length N, its periodicity condition $s_n = s_{n \mod N}$ leads to the modified definition of total variation:

$$\mathrm{TV}(\mathbf{s}) = \sum_{n=0}^{N-1} \left| s_n - s_{n-1 \mod N} \right|. \tag{8}$$

Using the time shift notation (5), we write (8) as

$$\mathrm{TV}(\mathbf{s}) = \left|\left| \mathbf{s} - \mathbf{C}_N \, \mathbf{s} \right|\right|_1. \tag{9}$$

For the circulant shift matrix (4), $|\lambda_{max}| = 1$ [9]. Hence, the definition (9) is conceptually an instantiation of the total variation on graphs (6) for finite discrete periodic time series. Similar to the way the total variation in time (8) measures a cumulative difference between signal values at connected nodes for the graph in Fig. 1, the total variation on graphs measures a cumulative difference between a signal value at each node and the values at its neighboring nodes for arbitrary graphs.

IV. Classification via Regularization

In classification, dataset elements are grouped in different classes by assigning a label to each element [3]. For example, the simplest case of binary classification considers only two classes. Respectively, labels can take only two different values, such as $+1$ and -1.

Consider a graph $G = (\mathcal{V}, \mathbf{A})$ with N vertices that represent a dataset with N elements. Each node corresponds to an element, and two nodes are connected if we know that the corresponding elements are similar to each other. If the connection is directed, the similarity is assumed to be known only in one direction.

Labels for this dataset form a signal indexed by the constructed graph. In particular, for a binary classification problem, known labels form the signal $\mathbf{s}^{(\text{known})}$ with values

$$s_n^{(\text{known})} = \begin{cases} +1, & n\text{th element belongs to class 1,} \\ -1, & n\text{th element belongs to class 2,} \\ 0, & \text{class is unknown.} \end{cases}$$

In the graph of dataset similarities, elements are connected if they are similar to each other. Since similar elements tend to belong to the same class, we expect that all labels for the dataset form a graph signal that does not change rapidly from node to node, i.e., signal values of connected nodes are likely to be the same. We formulate this assumption in terms of the total variation on graphs (6) by assuming that labels form a graph signal that has the lowest total variation [12]. That is, we define the predicted labels as the solution to the optimization problem

$$\mathbf{s}^{(\text{predicted})} = \underset{\mathbf{s} \in \mathbb{R}^N}{\mathbf{argmin}} \; \mathrm{TV_G}(\mathbf{s}). \tag{10}$$

Since the values of known labels should not change in the solution to (10), we also impose the condition that known labels are little changed from their original values in the predicted signal $\mathbf{s}^{(\text{predicted})}$. We write this requirement as

$$|| \, \mathbf{C} \, \mathbf{s}^{(\text{known})} - \mathbf{C} \, \mathbf{s} ||_2^2 < \epsilon, \tag{11}$$

where $\mathbf{C}$ is a $N \times N$ diagonal matrix such that

$$\mathbf{C}_{n,n} = \begin{cases} 1, & \text{if } s_n^{(\text{known})} \in \{+1, -1\}, \\ 0, & \text{otherwise.} \end{cases}$$

The parameter ϵ in the condition (11) controls how well known labels are preserved. It can be interpreted roughly as the number of known labels that can be changed to the opposite value by the minimization (10).

Once the predicted signal $\mathbf{s}^{(\text{predicted})}$ is calculated, the unlabeled data elements are assigned to one class if $s_n^{(\text{predicted})} > 0$ and another class otherwise.

Previous work. The problem of finding a signal with minimal variation under given conditions is called *regularization*.

Regularization based on time total variation (7) has been used for denoising, deblurring, and recovery of time series and digital images [14], [15]. A similar problem formulation has also been considered in data classification [16], [3], although it expressed the signal variation function using the Laplacian matrix. A Laplacian matrix is a $N \times N$ matrix

$$\mathbf{L} = \mathbf{D} - \mathbf{A},$$

where $\mathbf{D}$ is a diagonal matrix with elements given by

$$\mathbf{D}_{n,n} = \sum_{m=0}^{N-1} \mathbf{A}_{n,m}.$$

For consistency of notation, we write the signal variation function based on the Laplacian matrix as

$$\mathrm{TV}_{\mathrm{L}}(\mathbf{s}) = \mathbf{s}^T \, \mathbf{L} \, \mathbf{s} = \sum_{n,m} (s_n - s_m)^2 A_{n,m}. \qquad (12)$$

The function (12) was partially inspired by spectral graph theory [17], [16]. It has also been proposed recently for signal analysis on graphs [18], [19], [20], [21]. However, as we demonstrate in this paper, as well as in [12], it is sensitive to the quality of graphs that represent similarity of dataset elements and leads to lower classification accuracy in experiments with real-world datasets.

V. Experiments

This section illustrates the application of regularization on graphs (10) to multiclass classification. We have considered an example of binary classification in [12]. Here, we extend our approach to more than two classes. We study the performance of the regularization-based classification by comparing it with support vector machines and neural networks, which are well-known and widely used tools for feature-based classification [1], and regularization with Laplacian-based signal variation (12), which is a popular tool for classification based on similarity graphs.

A. Datasets and Graphs

We consider two large datasets: images of handwritten digits [22] and news postings [23].

The first dataset contains 70000 grayscale 28×28 images of handwritten digits from 0 to 9. We randomly select 3000 images of each digit and construct a testing dataset of 30000 images. The representation graph for this dataset is constructed by viewing each image as a point in a $28^2 = 784$-dimensional vector space, computing Euclidean distances between all images, and connecting each image with six nearest neighbors. When two images that correspond to nodes v_n and v_m are connected, we assign the weight

$$\mathbf{A}_{n,m} = e^{-d_{n,m}^2} \qquad (13)$$

to the corresponding edge, where $d_{n,m}$ is the Euclidean distance between the images.

The second dataset contains more than 18000 news articles on 20 different topics. We randomly select 500 articles on each topic to construct a testing dataset of 10000 articles. We represent each article with a vector that contains the number of occurrences of 6000 most common keywords, as described in [16]. The representation graph for this dataset is constructed by computing a cosine similarity measure between all keyword vectors, and connecting each article to six most similar articles. For two articles, the distance between them is calculated as the cosine of the angle between keyword vectors $\mathbf{v}_n$ and $\mathbf{v}_m$:

$$d_{n,m} = \frac{\langle \mathbf{v}_n, \mathbf{v}_m \rangle}{||\mathbf{v}_n||_2 \; ||\mathbf{v}_m||_2}. \qquad (14)$$

A smaller angle (14) implies more similarity between two articles. The corresponding edge weight is set to

$$\mathbf{A}_{n,m} = 1 - d_{n,m}. \qquad (15)$$

B. Methods

Section IV describes the solution the binary classification problem using regularization based on the graph shift (10). We extend this approach to multiclass problems by performing one-against-all classification for each class. This technique considers one class at a time and group all other classes into a second "super"-class. After solving (10) subject to the condition (11) for each class, we choose the most likely class for each dataset element. We set the value of the parameter ϵ to 1 in all experiments. This can be interpreted roughly as allowing at most one known label in the vector of predicted labels to change value to the opposite one.

For comparison, we also consider regularization with the Laplacian-based variation (12). We solve the minimization problem

$$\mathbf{s}^{(\text{predicted})} = \underset{\mathbf{s} \in \mathbb{R}^N}{\mathbf{argmin}} \; \mathrm{TV}_{\mathrm{L}}(\mathbf{s}) \qquad (16)$$

subject to the same condition (11) in the same series of one-against-all classifications as above. The value of ϵ is set to 1 in all experiments, as well.

We also compare our approach with support vector machines and neural networks [1]. These approaches do not use the graph of similarities. Rather, they rely on the analysis of *features* for dataset elements. In our experiments, such features are given by pixel intensities for images and by keyword occurrences for news articles. Since support vector machines are binary classifiers, we also run a series of one-against-all classifications to classify for each class. Neural networks can classify multiple classes at once, so we perform multiclass classification directly by constructing and training neural networks with one hidden layer of 30 nodes.

C. Results

For each dataset, we assume that between 0.5% and 30% of randomly selected labels are known initially. We measure the classification accuracy by calculating remaining labels and comparing them with the ground truth. Results of each test are averaged over 100 runs.

Average classification accuracies of images and news articles are shown in Fig. 2 and Fig. 3, respectively. In all experiments, the regularization based on the graph shift (10) achieves higher

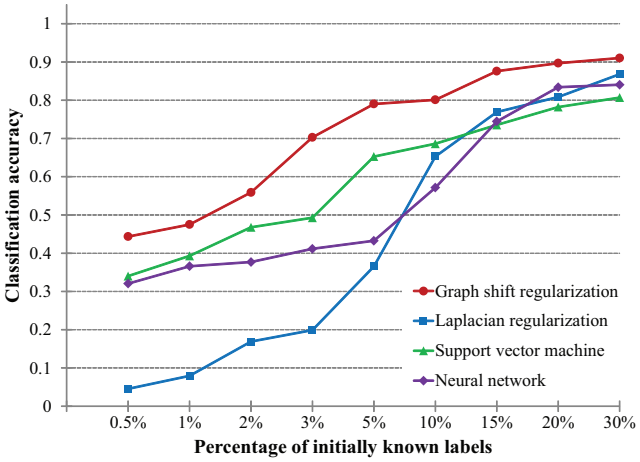

Fig. 2. Classification accuracy for the image dataset.

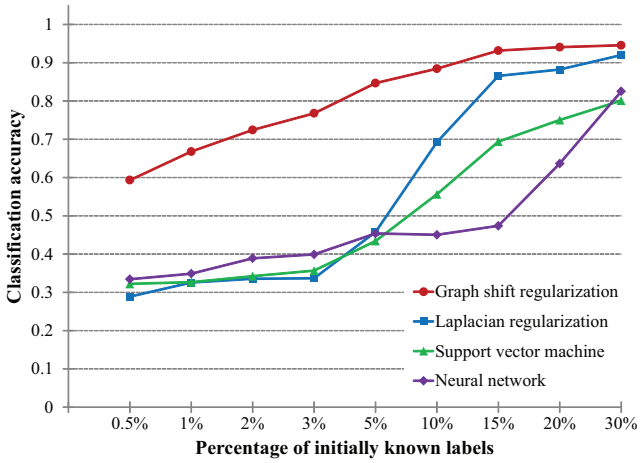

Fig. 3. Classification accuracy for the news articles dataset.

accuracy than other methods. For the images dataset, the accuracy of our approach for any fraction of initially known labels is between 5% and 20% higher then the closest competitor. For the news articles dataset, the gap in classification accuracies of our approach and other methods varies between 20% and as much as 40% when less than 15% of labels are known initially.

VI. Conclusions

We presented a classification algorithm that is based on the theory of discrete signal processing on graphs DSP_G. Our approach represents dataset elements and similarities between them using a graph and then views labels as a signal indexed by the nodes of the graph. We estimate unknown labels for dataset elements by searching for a smooth graph signal, the graph signal with lowest variation– a concept we introduce in [12]. We have demonstrated with experiments that the proposed approach yields high classification accuracy even for small fractions of initially known labels and that it outperforms other standard classification methods, including Laplacian-based regularization, support vector machines, and neural networks.

Acknowledgement

This work was supported in part by AFOSR grant FA95501210087.

References

[1] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, Wiley, 2nd edition, 2000.

[2] M. Belkin and P. Niyogi, "Semi-supervised learning on Riemannian manifolds," *Mach. Learn.*, vol. 56, no. 1–3, pp. 209–239, 2004.

[3] O. Chapelle, B. Schölkopf, and A. Zien, *Semi-Supervised Learning*, MIT Press, 2006.

[4] U. Luxburg, "A tutorial on spectral clustering," *Stat. Comput.*, vol. 17, no. 4, pp. 395–416, 2007.

[5] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs," *IEEE Trans. Signal Proc.*, vol. 61, no. 7, pp. 1644–1656, 2013.

[6] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs: Graph Fourier transform," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Proc.*, 2013.

[7] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs: Graph filters," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Proc.*, 2013.

[8] M. Püschel and J. M. F. Moura, "The algebraic approach to the discrete cosine and sine transforms and their fast algorithms," *SIAM J. Comp.*, vol. 32, no. 5, pp. 1280–1316, 2003.

[9] M. Püschel and J. M. F. Moura, "Algebraic signal processing theory: Foundation and 1-D time," *IEEE Trans. Signal Proc.*, vol. 56, no. 8, pp. 3572–3585, 2008.

[10] M. Püschel and J. M. F. Moura, "Algebraic signal processing theory: 1-D space," *IEEE Trans. Signal Proc.*, vol. 56, no. 8, pp. 3586–3599, 2008.

[11] A. Sandryhaila, J. Kovacevic, and M. Püschel, "Algebraic signal processing theory: 1-D Nearest-neighbor models," *IEEE Trans. on Signal Proc.*, vol. 60, no. 5, pp. 2247–2259, 2012.

[12] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs: Frequency analysis," *IEEE Trans. Signal Proc.*, June 2013, submitted for publication.

[13] M. Vetterli and J. Kovačević, *Wavelets and Subband Coding*, Signal Processing, Prentice Hall, Englewood Cliffs, NJ, 1995.

[14] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60, no. 1–4, pp. 259–268, 1992.

[15] T. F. Chan, S. Osher, and J. Shen, "The digital TV filter and nonlinear denoising," *IEEE Trans. Image Proc.*, vol. 10, no. 2, pp. 231–241, 2001.

[16] M. Belkin, I. Matveeva, and P. Niyogi, "Regularization and semi-supervised learning on large graphs," in *Proc. Conf. Learn. Th.*, 2004, pp. 624–638.

[17] F. R. K. Chung, *Spectral Graph Theory*, AMS, 1996.

[18] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *J. Appl. Comp. Harm. Anal.*, vol. 30, no. 2, pp. 129–150, 2011.

[19] A. Agaskar and Y. Lu, "A spectral graph uncertainty principle," *IEEE Trans. Inf. Theory*, vol. 59, no. 7, pp. 4338–4356, 2013.

[20] X. Zhu and M. Rabbat, "Approximating signals supported on graphs," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Proc.*, 2012, pp. 3921–3924.

[21] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs," *IEEE Signal Proc. Mag.*, vol. 30, no. 3, pp. 83–98, 2013.

[22] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[23] K. Lang, "NewsWeeder: Learning to filter netnews," in *Proc. Int. Conf. Mach. Learn.*, 1995, pp. 331–339.