THE COUNTERINTUITIVE MECHANISM OF GRAPH-BASED SEMI-SUPERVISED LEARNING IN THE BIG DATA REGIME

Xiaoyi MAI, Romain COUILLET

CentraleSupélec, Université Paris-Saclay, Gif-sur-Yvette, France

ABSTRACT

In this article, a new approach is proposed to study the performance of graph-based semi-supervised learning methods, under the assumptions that the dimension of data p and their number n grow large at the same rate and that the data arise from a Gaussian mixture model. Unlike small dimensional systems, the large dimensions allow for a Taylor expansion to linearize the weight (or kernel) matrix W, thereby providing in closed form the limiting performance of semi-supervised learning algorithms. This notably allows to predict the classification error rate as a function of the normalization parameters and of the choice of the kernel function. Despite the Gaussian assumption for the data, the theoretical findings match closely the performance achieved with real datasets, particularly here on the popular MNIST database.

Index Terms— semi-supervised learning, graphs, performance analysis, random matrix theory

1. INTRODUCTION

Semi-supervised learning is one of the important legs of machine learning research. It is of great practical interest to combine labeled data and unlabeled data, especially for the classification problems where the collection of labeled data is expensive and time-consuming.

Graph-based methods constitute one of the main branches of semi-supervised learning. Let $x_1, \ldots, x_n \in \mathbb{R}^p$ be data vectors. Among them are $n_{[l]}$ labeled data $x_1, \ldots, x_{n_l}$ and $n_{[u]} = n - n_{[l]}$ unlabeled data $x_{n_l+1}, \ldots, x_n$. Labeled and unlabeled data are considered as vertices in a weighted graph with edge weights W_{ij} reflecting the similarity between x_i and x_j, and defined through a kernel function f; here $W_{ij} = f(\|x_i - x_j\|^2/p)$. There are several approaches to consider the semi-supervised learning problem on a graph, such as label propagation, random walk, electrical network, etc. [1] Inspired by [2] which introduces an energy-based approach to classify the unlabeled data, we formulate here the problem in terms of the minimization of a cost function derived from the graph, with the presence of a normalization term α (as detailed in Section 2). The adjustment of α retrieves various known algorithms [3]. We propose here to identify the normalization which minimizes the classification error rate. The performance analysis is made difficult by the highly nonlinear structure of the algorithm and of W itself. To cope with this strong technical limitation, we exploit the approach developed in [4] to linearize the weight matrix W. The main idea of this approach is that, under appropriate assumptions on the data model, as n and p grow large at the same rate, the key value $\|x_i - x_j\|^2/p$ converges to a constant τ for all $i \neq j$, thereby allowing for the linearization of W_{ij} around $f(\tau)$. This linearization turns W into a tractable random matrix from which the (asymptotic) output of the classification algorithms can be statistically retrieved, analyzed, and improved over their latent parameters (kernel function choice, normalization parameters, etc.).

Our theoretical results lead to two crucial conclusions: (i) in the large dimensional regime, traditional graph-based semi-supervised learning methods, such as the Standard Laplacian method (see e.g., [2]) and the Normalized Laplacian method (see e.g., [5]) tend to fail, while the PageRank based method, which is identified in [3], is the only method to yield sensible classification results, and (ii) when the number of labeled data is not balanced for every class, the classification algorithm no longer works, but we provide a simple solution, consisting in adding a normalization step in the algorithm.

2. OPTIMIZATION FRAMEWORK

Let $x_1, \ldots, x_n \in \mathbb{R}^p$ be n vectors classified in K classes $\mathcal{C}_1, \ldots, \mathcal{C}_K$. The $n_{[l]}$ data $x_1, \ldots, x_{n_{[l]}}$ are *labeled* in the sense that the classes for these data are known, while the $n_{[u]} = n - n_{[l]}$ data $x_{n_{[l]}+1}, \ldots, x_n$ are kept *unlabeled*.

Following classical works [2, 3], we define an overall cost function.

$$\mathrm{C}(F) = \sum_{k=1}^{K} \sum_{1 \leq i,j \leq n} W_{ij} \|D_i^\alpha F_{ik} - D_j^\alpha F_{jk}\|^2$$

with F_{ik} containing the (soft if $F_{ik} \in \mathbb{R}$ or hard if $F_{ik} \in \{0, 1\}$) score for x_i to belong to class $\mathcal{C}_k$, $W_{ij} = f(\|x_i - x_j\|^2/p)$ for some kernel function f, $D_i = \sum_j W_{ij}$ and $\alpha \in \mathbb{R}$ a tuning parameter. In unsupervised learning, the value of

This work is supported by the ANR Project RMT4GRAPH (ANR-14-CE28-0006).

F is determined by minimizing C(F) [6]. In semi-supervised learning, since some data classes are known, we additionally impose the values of $F_{i,\cdot}$ for $1 \leq i \leq n_{[l]}$, thereby leading to the following constrained optimization problem:

$$\min_{F \in \mathbb{R}^{n \times K}} C(F) \ \text{ s.t.} \, F_{ik} = \delta_{x_i \in \mathcal{C}_k},\, 1 \leq i \leq n_{[l]},\, 1 \leq k \leq K \tag{1}$$

which has the explicit solution

$$F_{[u]} = (I_{n_u} - D_{[u]}^{-1-\alpha} W_{[uu]} D_{[u]}^{\alpha})^{-1} D_{[u]}^{-1-\alpha} W_{[ul]} D_{[l]}^{\alpha} F_{[l]} \tag{2}$$

where $F = \begin{bmatrix} F_{[l]} \\ F_{[u]} \end{bmatrix}$, $W = \{W_{ij}\}_{i,j=1}^{n} = \begin{bmatrix} W_{[ll]} & W_{[lu]} \\ W_{[ul]} & W_{[uu]} \end{bmatrix}$ and $D = \operatorname{diag}(\{D_i\}) = \begin{bmatrix} D_{[l]} & 0 \\ 0 & D_{[u]} \end{bmatrix}$. The final (hard) decision consists in classifying x_i in $\mathcal{C}_a$ for $a = \max_{k \in [1,\ldots,K]} F_{ik}$.

The performance of the classification algorithm can be highly dependent on the value of α. In the literature, there are two common choices for α: $\alpha = 0$ corresponding to the Standard Laplacian method, $\alpha = -\frac{1}{2}$ corresponding to the Normalized Laplacian method. A third choice, $\alpha = -1$, is discussed and named PageRank based method in [3].

3. MODEL AND THEORETICAL RESULTS

3.1. Model and assumptions

To obtain quantitative classification performance after the hard-decision step following (2), we assume that the data x_i are retrieved from a mixture of K Gaussian (corresponding to the classes) and let $n, p \to \infty$. Specifically, for $k \in \{1, \ldots, K\}$, $x_i \in \mathcal{C}_k$ corresponds to $x_i \sim \mathcal{N}(\mu_k, C_k)$. There are n_k instances in $\mathcal{C}_k$, among which $n_{[l]k}$ are labeled and $n_{[u]k}$ are unlabeled.

When the number of data points n and their dimension p grow simultaneously large, in order to ensure that the classification problem we study here is non trivial, in the sense that the asymptotic error rate is neither 0 nor 1, μ_k and C_k are chosen to behave in a prescribed manner, as described in [4] for the unsupervised scenario.[1]

Assumption 1 (Growth Rate)**.** *As $n \to \infty$, $\frac{p}{n} \to c_0 > 0$, $\frac{n_k}{n} \to c_k > 0$, and $\frac{n_{[l]}}{n} \to c_{[l]} > 0$. Besides,*

1. *For $\mu^o \triangleq \sum_{k=1}^{K} \frac{n_k}{n} \mu_k$ and $\mu_k^o \triangleq \mu_k - \mu^o$, $\|\mu_k^o\| = O(1)$.*
2. *For $C^o \triangleq \sum_{k=1}^{K} \frac{n_k}{n} C_k$ and $C_k^o \triangleq C_k - C^o$, $\|C_k\| = O(1)$ and $\operatorname{tr} C_k^o = O(\sqrt{n})$.*
3. *As $n \to \infty$, $\frac{2}{p} \operatorname{tr} C^o \to \tau$.*

As for the kernel function, it follows the assumption below.

Assumption 2 (Kernel function)**.** *The kernel function $f : \mathbb{R}^+ \to \mathbb{R}^+$ is three-times continuously differentiable in a neighborhood of τ.*

The constant τ introduced in the third point of Assumption 1 is important as $\|x_i - x_j\|^2/p \to \tau$ almost surely and uniformly on all $i \neq j \in \{1, \ldots, n\}$, which implies that all data points x_i are almost equally far away from each other whether or not they belong to the same class. As a consequence, the W_{ij}'s, which are supposed to reflect the similarity between x_i's , are asymptotically the same for all $i \neq j$, thus do not exert its expected effect in the cost function $\mathrm{C}(F)$, as they do for small dimensional data. For that reason, we do not expect the classification algorithm to work in our case, but contrary to our intuition, it shall in fact work if and only if $\alpha \simeq -1$, as explained in the subsection below.

3.2. Principle and results

As $\|x_i - x_j\|^2/p \to \tau$ almost surely, the off-diagonal entries of W can be Taylor-expanded around $f(\tau)$, which allows for the decomposition of W and D into a series of tractable random matrices. However, to access $F_{[u]}$ in (2), this leaves us with the complicated inverse of the matrix $I_{n_u} - D_{[u]}^{-1-\alpha} W_{[uu]} D_{[u]}^{\alpha}$ to handle. Fortunately, after decomposition, we find that

$$W_{[uu]} = f(\tau) 1_{n_{[u]}} 1_{n_{[u]}}^T + O(n^{\frac{1}{2}})$$
$$D_{[u]} = n f(\tau) I_{n_{[u]}} + O(n^{\frac{1}{2}})$$

where $O(\cdot)$ is with respect to the matrix operator norm, and similarly for $W_{[ul]}$ and $D_{[l]}$, which then leads to

$$D_{[u]}^{-1-\alpha} W_{[uu]} D_{[u]}^{\alpha} = \frac{1}{n} 1_{n_{[u]}} 1_{n_{[u]}}^T + O(n^{-\frac{1}{2}}).$$

Therefore, $(I_{n_{[u]}} - D_{[u]}^{-1-\alpha} W_{[uu]} D_{[u]}^{\alpha})^{-1}$ can be Taylor-expanded around

$$\left(I_{n_{[u]}} - \frac{1}{n} 1_{n_{[u]}} 1_{n_{[u]}}^T \right)^{-1} = I_{n_{[u]}} + \frac{1}{n_{[l]}} 1_{n_{[u]}} 1_{n_{[u]}}^T.$$

After explicit computation of the various $O(\cdot)$ terms above (the full derivation is left to an extended version of the article), we are able to analyze (2).

Proposition 1. *The columns of $F_{[u]}$ can be expressed as*[2]

$$(F_{[u]})_{\cdot a} = \frac{n_{[l]a}}{n} \left[\underbrace{v}_{O(1)} + (1+\alpha) \underbrace{\kappa \frac{\operatorname{tr} C_a^o}{p} 1_{n_{[u]}}}_{O(n^{-\frac{1}{2}})} \right] + O(n^{-1}) \tag{3}$$

for some constant κ and a random vector v, both independent of a and of entries of order $O(1)$. Here 1_n is the vector of ones of size n.

[1] Since semi-supervised learning is simpler than unsupervised learning, the algorithms ought to perform at least as well under these conditions.

[2] The $O(\cdot)$ terms below are understood entry-wise.

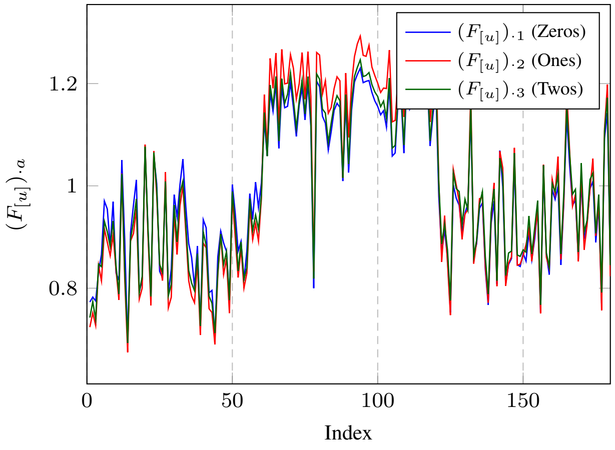

Fig. 1. Vectors $(F_{[u]})_{\cdot a}$ for 3-class MNIST data (zeros, ones, twos), $n = 192$, $p = 784$, $n_{[l]}/n = 1/16$, $n_{[l]1} = n_{[l]2} = n_{[l]3}$, $n_{[u]1} = n_{[u]2} = n_{[u]3}$, Gaussian kernel, $\alpha = -1$.

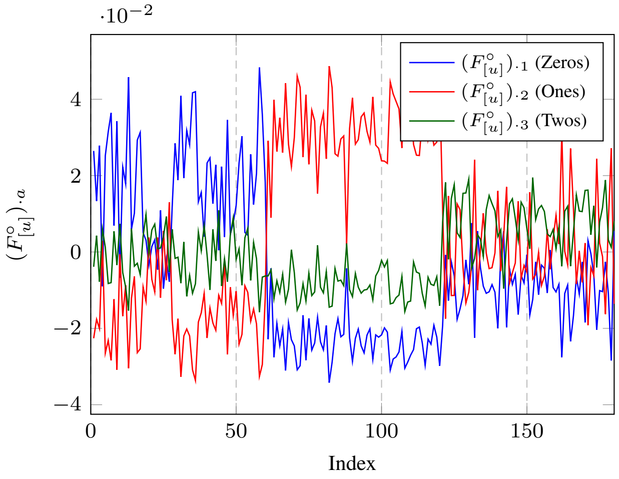

Fig. 2. Centered vectors $(F^{\circ}_{[u]})_{\cdot a} = (F_{[u]})_{\cdot a} - \sum_{b=1}^{K} \frac{n_{[l]b}}{n_{[l]}}(F_{[u]})_{\cdot b}$ for 3-class MNIST data (zeros, ones, twos), $n = 192, p = 784, n_l/n = 1/16, n_{[l]1} = n_{[l]2} = n_{[l]3}$, $n_{[u]1} = n_{[u]2} = n_{[u]3}$, Gaussian kernel, $\alpha = -1$.

The two curly-bracketed terms in the right-hand side of (3) do not contain relevant information about the class of unlabeled points, which is solely contained within the tailing term $O(n^{-1})$ (see the extended version of the article for a full characterization of v). More importantly, they are fatally harmful to the classification as the first term induces strong bias of higher order than the information terms when $n_{[l]a}$ is not the same for all $a \in \{1, \cdots, K\}$ and so does the second term when we do not have $\alpha \simeq -1$. Our first conclusions are then that, for the classification algorithm to behave correctly in the large dimensional regime, we need to:

(i) Adapt the hard decision step from comparing $(F_{[u]})_{\cdot a}$ to comparing $(F_{[u]})_{\cdot a}/n_{[l]a}$ so as to eliminate the bias induced by the first term (by making it the same for every column of $F_{[u]}$);

(ii) Impose $\alpha = -1 + O(n^{-\frac{1}{2}})$ in order to degrade the bias due to the second term.

As observed in Figure 1, $(F_{[u]})_{\cdot a}$ are indistinguishable at the first order for MNIST data. This is unsettling at first sight because $(F_{[u]})_{\cdot a}$ do not behave at all like $(F_{[l]})_{\cdot a}$, which is the purpose of the algorithm. This unexpected behavior of $(F_{[u]})_{\cdot a}$ is again due to the large dimensional effect of data and is consistent with our analysis. A detailed calculus in fact reveals that the classification algorithm works nonetheless with $\alpha \simeq -1$ thanks to the information terms contained within the tailing term $O(n^{-1})$ of (3), which allow for the separation of classes, as shown in Figure 2.

With these remarks in mind and with some further calculus (available in the extended version of the article), we are in position to provide the main result of the article:

Theorem 1. *For $x_i \in \mathcal{C}_b$ unlabeled, let $\hat{F}_{ia} = \frac{np}{n_{[l]a}} F_{ia}$ and $\alpha = -1 + \frac{\beta}{\sqrt{p}}$. Then, under Assumptions 1–2, $\hat{F}_{i.} - G_b \to 0$ weakly, with $G_b \sim \mathcal{N}(M_b, \Sigma_b)$, where*

$$
\begin{aligned}
(M_b)_a \quad &= \quad -\frac{2f'(\tau)}{f(\tau)}\tilde{\mu^o_a}\tilde{\mu^o_b} + \frac{f^{''}(\tau)}{f(\tau)}\frac{\mathrm{tr}\tilde{C^o_a}}{\sqrt{p}}\frac{\mathrm{tr}\tilde{C^o_b}}{\sqrt{p}} \\
&\quad + \frac{2f^{''}(\tau)}{f(\tau)}\frac{\mathrm{tr}(\tilde{C}_a\tilde{C}_b)}{p} - \frac{f^{'}(\tau)^2}{f(\tau)^2}\frac{\mathrm{tr}C^o_a}{\sqrt{p}}\frac{\mathrm{tr}C^o_b}{\sqrt{p}} \\
&\quad + \frac{n\beta}{n_{[l]}}\frac{f^{'}(\tau)}{f(\tau)}\frac{\mathrm{tr}C^o_a}{\sqrt{p}} + B_b \qquad (4)
\end{aligned}
$$

$$
\begin{aligned}
(\Sigma_b)_{a_1a_2} \quad &= \quad \left(\frac{f^{''}(\tau)}{f(\tau)} - \frac{f^{'}(\tau)^2}{f(\tau)^2}\right)^2 \frac{2\mathrm{tr}C_b^2\mathrm{tr}C^o_{a_1}\mathrm{tr}C^o_{a_2}}{p^2} \\
&\quad + \frac{4f^{'}(\tau)^2}{f(\tau)^2}\left[\mu^{oT}_{a_1}C_b\mu^o_{a_2} + \delta^{a_2}_{a_1}\frac{\mathrm{tr}C_bC_{a_1}}{n_{[l]a_1}}\right] \quad (5)
\end{aligned}
$$

where B_b is a constant bias of order $O(n)$, $\tilde{\mu^o_a} = \mu^o_a - \sum_{d=1}^{K}\frac{n_{[l]d}}{n_{[l]}}\mu^o_d$, $\tilde{C^o_a} = C^o_a - \sum_{d=1}^{K}\frac{n_{[l]d}}{n_{[l]}}C^o_d$ and $\tilde{C}_a = C_a - \sum_{d=1}^{K}\frac{n_{[l]d}}{n_{[l]}}C_d$.

Since $\hat{F}_{i.}$ is asymptotically a Gaussian vector, we easily access the asymptotic misclassification rate for the unlabeled data. In particular, for $K = 2$, we have:

Corollary 1. *Under the conditions of Theorem 1, and with $K = 2$, we have, for $a \neq b \in \{1, 2\}$,*

$$
P(x_i \to \mathcal{C}_a | x_i \in \mathcal{C}_b) - Q\left(\frac{(M_b)_b - (M_b)_a}{\sqrt{j^T\Sigma_b j}}\right) \to 0 \qquad (6)
$$

where $j = [1 - 1]^T$ and $Q(x) = \frac{1}{2\pi}\int_x^\infty \exp(-t^2/2)dt$.

From (4) and (5), we see that $f(\tau)$, $f^{'}(\tau)$, $f^{''}(\tau)$ and β are the parameters influencing the output of the classification algorithm. When $K = 2$, some interesting conclusions for these parameters are readily drawn. From Corollary 1, we deduce that for the classification algorithm to perform better than random, we need to ensure that $(M_b)_b - (M_b)_a > 0$.

Keeping this in mind, we now take a closer look at each term of (4):

(i) Evidently, $\tilde{\mu_b^o}\tilde{\mu_b^o} \geq \tilde{\mu_a^o}\tilde{\mu_b^o}$. So for the the first term, in order to have $-\frac{2f'(\tau)}{f(\tau)}\tilde{\mu_b^o}\tilde{\mu_b^o} \geq -\frac{2f'(\tau)}{f(\tau)}\tilde{\mu_a^o}\tilde{\mu_b^o}$, we need $f^{'}(\tau) < 0$ since $f(\tau) > 0$ according to Assumption 2. The same reasoning for the second and third terms induces $f^{''}(\tau) > 0$.

(ii) Since $\mathrm{tr}C_b^o\mathrm{tr}C_b^o \geq \mathrm{tr}C_a^o\mathrm{tr}C_b^o$, for the forth term, we always have $-\frac{f^{'}(\tau)^2}{npf(\tau)^2}\frac{\mathrm{tr}C_b^o}{\sqrt{p}}\frac{\mathrm{tr}C_b^o}{\sqrt{p}} \geq -\frac{f^{'}(\tau)^2}{npf(\tau)^2}\frac{\mathrm{tr}C_a^o}{\sqrt{p}}\frac{\mathrm{tr}C_b^o}{\sqrt{p}}$ for any $f'(\tau)$. So this term is harmful, working in the opposite direction to $(M_b)_b - (M_b)_a > 0$.

(iii) For the fifth term, we have $\frac{n\beta}{n_{[l]}}\frac{f^{'}(\tau)}{f(\tau)}\frac{\mathrm{tr}C_b^o - \mathrm{tr}C_a^o}{\sqrt{p}}$ for $(M_b)_b - (M_b)_a$ and $\frac{n\beta}{n_{[l]}}\frac{f^{'}(\tau)}{f(\tau)}\frac{\mathrm{tr}C_a^o - \mathrm{tr}C_b^o}{\sqrt{p}} = -\frac{n\beta}{n_{[l]}}\frac{f^{'}(\tau)}{f(\tau)}\frac{\mathrm{tr}C_b^o - \mathrm{tr}C_a^o}{\sqrt{p}}$ for $(M_a)_a - (M_a)_b$. It is a balance term in the sense that, if it increases $(M_b)_b - (M_b)_a$, it decreases inevitably $(M_a)_a - (M_a)_b$. As it is the only term containing β in (4) and (5), it allows to decrease $P(x_i \to \mathcal{C}_a | x_i \in \mathcal{C}_b)$ at the expense of $P(x_i \to \mathcal{C}_b | x_i \in \mathcal{C}_a)$ or the other way around through the adjustment of β.

4. SIMULATIONS

We provide in this section two simulations conducted respectively on Gaussian data as defined in Assumption 1 and on the real-world MNIST database [7]. Here, for the theoretical comparison, we handle the MNIST data as if they were Gaussian with means and covariances empirically obtained from the full set of 13 007 images. In those simulations, we display the actual classification accuracy as a function of α, and the theoretical curve obtained by applying Corollary 1. We find that our theoretical results match extremely well the reality not only for Gaussian data (Figure 3), but, quite surprisingly, also for MNIST data (Figure 4) which are evidently not Gaussian. This suggests that most of the image information is retrieved from its first order (empirical) moments.

5. CONCLUDING REMARKS

This article has proposed a novel (random matrix-based) framework of semi-supervised learning analysis. The linearization of (2) led to theoretical results providing direct access to the performance of the classification algorithm in the large data regime. The high consistence of our model with MNIST data suggests the appropriateness of a mere Gaussian modeling in the large dimensional data regime. From the theoretical results, we deduced several important conclusions concerning the algorithm: (i) its outcome is strongly biased by $n_{[l]a}$ unless we change the algorithm from comparing $(F_{[u]})_{\cdot a}$ to comparing $(F_{[u]})_{\cdot a}/n_{[l]a}$. (ii) α should be taken around -1, which implies that among three existing methods in the literature, the PageRank method is the only one that works in our setting of large dimensional data. (iii) When $K = 2$, f should be chosen so that $f^{'}(\tau) < 0$, $f^{''}(\tau) > 0$ in order to ensure the good working of the algorithm.

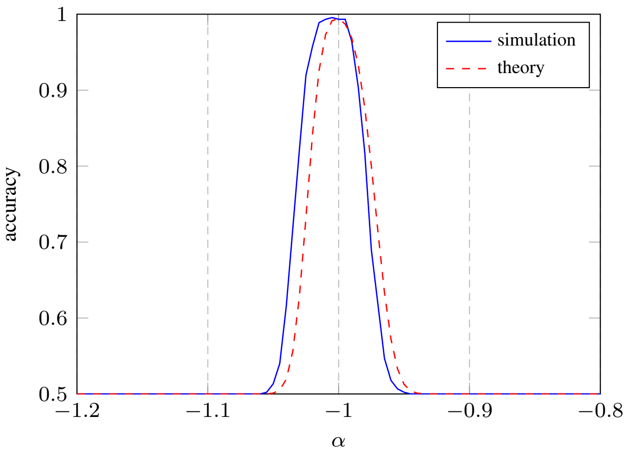

Fig. 3. Theoretical and empirical accuracy as a function of α for 2-class Gaussian data, $n = 512$, $p = 1024$, $n_{[l]}/n = 1/8$, $n_{[u]1} = n_{[u]2}$, Gaussian kernel.

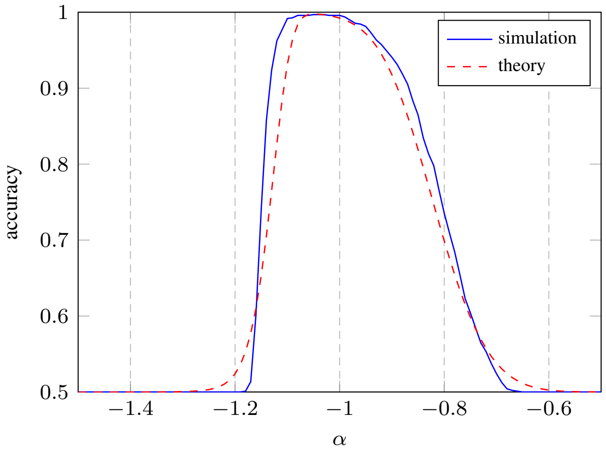

Fig. 4. Theoretical and empirical accuracy as a function of α for 2-class MNIST data (zeros, ones), $n = 1024$, $p = 784$, $n_{[l]}/n = 1/16$, $n_{[u]1} = n_{[u]2}$, Gaussian kernel.

An important extension of this work would be to handle discrete kernel matrices such as the popular k-nearest neighbor approach. More importantly, the advent of new semi-supervised learning approaches arising from the sparse big data [8] and signal processing on graph [9, 10] fields calls for a development of our random matrix tools to encompass these novel approaches.

6. REFERENCES

[1] Xiaojin Zhu, John Lafferty, and Ronald Rosenfeld, *Semi-supervised learning with graphs*, Carnegie Mellon University, language technologies institute, school of computer science, 2005.

[2] Xiaojin Zhu, Zoubin Ghahramani, John Lafferty, et al., "Semi-supervised learning using gaussian fields and harmonic functions," in *ICML*, 2003, vol. 3, pp. 912–919.

[3] Konstantin Avrachenkov, Paulo Gonçalves, Alexey Mishenin, and Marina Sokol, "Generalized optimization framework for graph-based semi-supervised learning," *arXiv preprint arXiv:1110.4278*, 2011.

[4] Romain Couillet and Florent Benaych-Georges, "Kernel spectral clustering of large dimensional data," *arXiv preprint arXiv:1510.03547*, 2015.

[5] Dengyong Zhou, Olivier Bousquet, Thomas Navin Lal, Jason Weston, and Bernhard Schölkopf, "Learning with local and global consistency," *Advances in neural information processing systems*, vol. 16, no. 16, pp. 321–328, 2004.

[6] Ulrike Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[7] Yann LeCun, Corinna Cortes, and Christopher JC Burges, "The mnist database of handwritten digits," 1998.

[8] Mahdi Soltanolkotabi, Ehsan Elhamifar, Emmanuel J Candes, et al., "Robust subspace clustering," *The Annals of Statistics*, vol. 42, no. 2, pp. 669–699, 2014.

[9] Aliaksei Sandryhaila and José MF Moura, "Discrete signal processing on graphs," *IEEE transactions on signal processing*, vol. 61, no. 7, pp. 1644–1656, 2013.

[10] Aliaksei Sandryhaila and Jose MF Moura, "Discrete signal processing on graphs: Frequency analysis," *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3042–3054, 2014.