Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions

Xiaojin Zhu* ZHUXJ@CS.CMU.EDU
Zoubin Ghahramani†* ZOUBIN@GATSBY.UCL.AC.UK
John Lafferty* LAFFERTY@CS.CMU.EDU
*School of Computer Science, Carnegie Mellon University, Pittsburgh PA 15213, USA
†Gatsby Computational Neuroscience Unit, University College London, London WC1N 3AR, UK

Abstract

An approach to semi-supervised learning is proposed that is based on a Gaussian random field model. Labeled and unlabeled data are represented as vertices in a weighted graph, with edge weights encoding the similarity between instances. The learning problem is then formulated in terms of a Gaussian random field on this graph, where the mean of the field is characterized in terms of harmonic functions, and is efficiently obtained using matrix methods or belief propagation. The resulting learning algorithms have intimate connections with random walks, electric networks, and spectral graph theory. We discuss methods to incorporate class priors and the predictions of classifiers obtained by supervised learning. We also propose a method of parameter learning by entropy minimization, and show the algorithm's ability to perform feature selection. Promising experimental results are presented for synthetic data, digit classification, and text classification tasks.

1. Introduction

In many traditional approaches to machine learning, a target function is estimated using labeled data, which can be thought of as examples given by a "teacher" to a "student." Labeled examples are often, however, very time consuming and expensive to obtain, as they require the efforts of human annotators, who must often be quite skilled. For instance, obtaining a single labeled example for protein shape classification, which is one of the grand challenges of biological and computational science, requires months of expensive analysis by expert crystallographers. The problem of effectively combining *unlabeled* data with labeled data is therefore of central importance in machine learning.

The semi-supervised learning problem has attracted an increasing amount of interest recently, and several novel approaches have been proposed; we refer to (Seeger, 2001) for an overview. Among these methods is a promising family of techniques that exploit the "manifold structure" of the data; such methods are generally based upon an assumption that similar unlabeled examples should be given the same classification. In this paper we introduce a new approach to semi-supervised learning that is based on a random field model defined on a weighted graph over the unlabeled and labeled data, where the weights are given in terms of a similarity function between instances.

Unlike other recent work based on energy minimization and random fields in machine learning (Blum & Chawla, 2001) and image processing (Boykov et al., 2001), we adopt *Gaussian* fields over a continuous state space rather than random fields over the discrete label set. This "relaxation" to a continuous rather than discrete sample space results in many attractive properties. In particular, the most probable configuration of the field is unique, is characterized in terms of harmonic functions, and has a closed form solution that can be computed using matrix methods or loopy belief propagation (Weiss et al., 2001). In contrast, for multi-label discrete random fields, computing the lowest energy configuration is typically NP-hard, and approximation algorithms or other heuristics must be used (Boykov et al., 2001). The resulting classification algorithms for Gaussian fields can be viewed as a form of nearest neighbor approach, where the nearest labeled examples are computed in terms of a random walk on the graph. The learning methods introduced here have intimate connections with random walks, electric networks, and spectral graph theory, in particular heat kernels and normalized cuts.

In our basic approach the solution is solely based on the structure of the data manifold, which is derived from data features. In practice, however, this derived manifold structure may be insufficient for accurate classification. We

Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003), Washington DC, 2003.

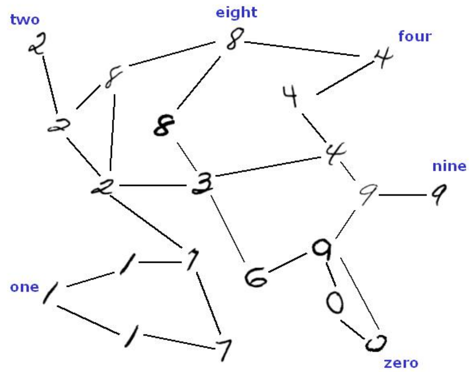

Figure 1. The random fields used in this work are constructed on labeled and unlabeled examples. We form a graph with weighted edges between instances (in this case scanned digits), with labeled data items appearing as special "boundary" points, and unlabeled points as "interior" points. We consider Gaussian random fields on this graph.

show how the extra evidence of class priors can help classification in Section 4. Alternatively, we may combine external classifiers using vertex weights or "assignment costs," as described in Section 5. Encouraging experimental results for synthetic data, digit classification, and text classification tasks are presented in Section 7. One difficulty with the random field approach is that the right choice of graph is often not entirely clear, and it may be desirable to learn it from data. In Section 6 we propose a method for learning these weights by entropy minimization, and show the algorithm's ability to perform feature selection to better characterize the data manifold.

2. Basic Framework

We suppose there are l labeled points $(x_1, y_1), \ldots, (x_l, y_l)$, and u unlabeled points $x_{l+1}, \ldots, x_{l+u}$; typically $l \ll u$. Let $n = l + u$ be the total number of data points. To begin, we assume the labels are binary: $y \in \{0, 1\}$. Consider a connected graph $G = (V, E)$ with nodes V corresponding to the n data points, with nodes $L = \{1, \ldots, l\}$ corresponding to the labeled points with labels $y_1, \ldots, y_l$, and nodes $U = \{l + 1, \ldots, l + u\}$ corresponding to the unlabeled points. Our task is to assign labels to nodes U. We assume an $n \times n$ symmetric weight matrix W on the edges of the graph is given. For example, when $x \in \mathbb{R}^m$, the weight matrix can be

$$w_{ij} = \exp\left(-\sum_{d=1}^{m} \frac{(x_{id} - x_{jd})^2}{\sigma_d^2}\right) \tag{1}$$

where x_{id} is the d-th component of instance x_i represented as a vector $x_i \in \mathbb{R}^m$, and $\sigma_1, \ldots, \sigma_m$ are length scale hyperparameters for each dimension. Thus, nearby points in Euclidean space are assigned large edge weight. Other weightings are possible, of course, and may be more appropriate when x is discrete or symbolic. For our purposes the matrix W fully specifies the data manifold structure (see Figure 1).

Our strategy is to first compute a *real-valued* function $f : V \longrightarrow \mathbb{R}$ on G with certain nice properties, and to then assign labels based on f. We constrain f to take values $f(i) = f_l(i) \equiv y_i$ on the labeled data $i = 1, \ldots, l$. Intuitively, we want unlabeled points that are nearby in the graph to have similar labels. This motivates the choice of the quadratic energy function

$$E(f) = \frac{1}{2} \sum_{i,j} w_{ij} \left(f(i) - f(j)\right)^2 \tag{2}$$

To assign a probability distribution on functions f, we form the Gaussian field $p_\beta(f) = \frac{e^{-\beta E(f)}}{Z_\beta}$, where β is an "inverse temperature" parameter, and Z_β is the partition function $Z_\beta = \int_{f|_L = f_l} \exp\left(-\beta E(f)\right) df$, which normalizes over all functions constrained to f_l on the labeled data.

It is not difficult to show that the minimum energy function $f = \arg\min_{f|_L = f_l} E(f)$ is *harmonic*; namely, it satisfies $\Delta f = 0$ on unlabeled data points U, and is equal to f_l on the labeled data points L. Here Δ is the *combinatorial Laplacian*, given in matrix form as $\Delta = D - W$ where $D = \text{diag}(d_i)$ is the diagonal matrix with entries $d_i = \sum_j w_{ij}$ and $W = [w_{ij}]$ is the weight matrix.

The harmonic property means that the value of f at each unlabeled data point is the average of f at neighboring points:

$$f(j) = \frac{1}{d_j} \sum_{i \sim j} w_{ij} f(i), \text{ for } j = l + 1, \ldots, l + u \tag{3}$$

which is consistent with our prior notion of smoothness of f with respect to the graph. Expressed slightly differently, $f = Pf$, where $P = D^{-1}W$. Because of the maximum principle of harmonic functions (Doyle & Snell, 1984), f is unique and is either a constant or it satisfies $0 < f(j) < 1$ for $j \in U$.

To compute the harmonic solution explicitly in terms of matrix operations, we split the weight matrix W (and similarly D, P) into 4 blocks after the lth row and column:

$$W = \begin{bmatrix} W_{ll} & W_{lu} \\ W_{ul} & W_{uu} \end{bmatrix} \tag{4}$$

Letting $f = \begin{bmatrix} f_l \\ f_u \end{bmatrix}$ where f_u denotes the values on the unlabeled data points, the harmonic solution $\Delta f = 0$ subject to $f|_L = f_l$ is given by

$$f_u = (D_{uu} - W_{uu})^{-1} W_{ul} f_l = (I - P_{uu})^{-1} P_{ul} f_l \tag{5}$$

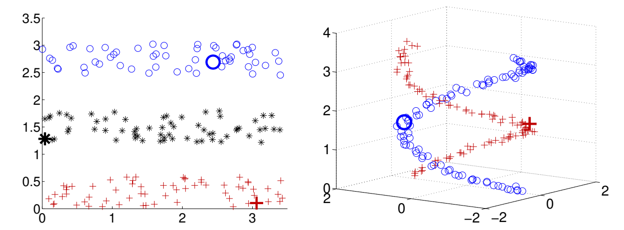

Figure 2. Demonstration of harmonic energy minimization on two synthetic datasets. Large symbols indicate labeled data, other points are unlabeled.

In this paper we focus on the above harmonic function as a basis for semi-supervised classification. However, we emphasize that the Gaussian random field model from which this function is derived provides the learning framework with a consistent probabilistic semantics.

In the following, we refer to the procedure described above as *harmonic energy minimization*, to underscore the harmonic property (3) as well as the objective function being minimized. Figure 2 demonstrates the use of harmonic energy minimization on two synthetic datasets. The left figure shows that the data has three bands, with $l = 3$, $u = 178$, and $\sigma = 0.22$; the right figure shows two spirals, with $l = 2$, $u = 184$, and $\sigma = 0.43$. Here we see harmonic energy minimization clearly follows the structure of data, while obviously methods such as kNN would fail to do so.

3. Interpretation and Connections

As outlined briefly in this section, the basic framework presented in the previous section can be viewed in several fundamentally different ways, and these different viewpoints provide a rich and complementary set of techniques for reasoning about this approach to the semi-supervised learning problem.

3.1. Random Walks and Electric Networks

Imagine a particle walking along the graph G. Starting from an unlabeled node i, it moves to a node j with probability P_{ij} after one step. The walk continues until the particle hits a labeled node. Then $f(i)$ is the probability that the particle, starting from node i, hits a labeled node with label 1. Here the labeled data is viewed as an "absorbing boundary" for the random walk.

This view of the harmonic solution indicates that it is closely related to the random walk approach of Szummer and Jaakkola (2001), however there are two major differences. First, we fix the value of f on the labeled points, and second, our solution is an equilibrium state, expressed in terms of a hitting time, while in (Szummer & Jaakkola, 2001) the walk crucially depends on the time parameter t. We will return to this point when discussing heat kernels.

An electrical network interpretation is given in (Doyle & Snell, 1984). Imagine the edges of G to be resistors with conductance W. We connect nodes labeled 1 to a positive voltage source, and points labeled 0 to ground. Then f_u is the voltage in the resulting electric network on each of the unlabeled nodes. Furthermore f_u minimizes the energy dissipation of the electric network G for the given f_l. The harmonic property here follows from Kirchoff's and Ohm's laws, and the maximum principle then shows that this is precisely the same solution obtained in (5).

3.2. Graph Kernels

The solution f can be viewed from the viewpoint of spectral graph theory. The heat kernel with time parameter t on the graph G is defined as $K_t = e^{-t\Delta}$. Here $K_t(i,j)$ is the solution to the heat equation on the graph with initial conditions being a point source at i at time $t = 0$. Kondor and Lafferty (2002) propose this as an appropriate kernel for machine learning with categorical data. When used in a kernel method such as a support vector machine, the kernel classifier $\hat{f}_t(j) = \sum_{i \in L} \alpha_i y_i K_t(i,j)$ can be viewed as a solution to the heat equation with initial heat sources $\alpha_i y_i$ on the labeled data. The time parameter t must, however, be chosen using an auxiliary technique, for example cross-validation.

Our algorithm uses a different approach which is *independent* of t, the diffusion time. Let Δ_{uu} be the lower right $u \times u$ submatrix of Δ. Since $\Delta_{uu} = D_{uu} - W_{uu}$, it is the Laplacian restricted to the unlabeled nodes in G. Consider the heat kernel on this submatrix: $K_t^{'} = e^{-t\Delta_{uu}}$. Then $K_t^{'}$ describes heat diffusion on the unlabeled subgraph with Dirichlet boundary conditions on the labeled nodes. The *Green's function* $\mathcal{G}$ is the inverse operator of the restricted Laplacian, $\mathcal{G}\Delta_{uu} = I$, which can be expressed in terms of the integral over time of the heat kernel $K_t^{'}$:

$$\mathcal{G} = \int_0^\infty K_t^{'} dt = \int_0^\infty e^{-t\Delta_{uu}} dt = (D_{uu} - W_{uu})^{-1} \quad (6)$$

The harmonic solution (5) can then be written as

$$f_u = \mathcal{G} W_{ul} f_l \quad \text{or} \quad f(j) = \sum_{i=1}^{l} \sum_k y_i \, w_{ik} \, \mathcal{G}(k,j) \quad (7)$$

Expression (7) shows that this approach can be viewed as a kernel classifier with the kernel $\mathcal{G}$ and a specific form of kernel machine. (See also (Chung & Yau, 2000), where a normalized Laplacian is used instead of the combinatorial Laplacian.) From (6) we also see that the spectrum of $\mathcal{G}$ is $\{\lambda_i^{-1}\}$, where $\{\lambda_i\}$ is the spectrum of Δ_{uu}. This indicates a connection to the work of Chapelle et al. (2002), who manipulate the eigenvalues of the Laplacian to create various kernels. A related approach is given by Belkin and Niyogi (2002), who propose to regularize functions on G by selecting the top p normalized eigenvectors of Δ corresponding to the smallest eigenvalues, thus obtaining the best fit to f_l in the least squares sense. We remark that our f fits the labeled data exactly, while the order p approximation may not.

3.3. Spectral Clustering and Graph Mincuts

The normalized cut approach of Shi and Malik (2000) has as its objective function the minimization of the Raleigh quotient

$$R(f) = \frac{f^\top \Delta f}{f^\top D f} = \frac{\sum_{ij} w_{ij}(f(i) - f(j))^2}{\sum_i d_i f(i)^2} \qquad (8)$$

subject to the constraint $f \perp \mathbf{1}$. The solution is the second smallest eigenvector of the generalized eigenvalue problem $\Delta f = \lambda D f$. Yu and Shi (2001) add a grouping bias to the normalized cut to specify which points should be in the same group. Since labeled data can be encoded into such pairwise grouping constraints, this technique can be applied to semi-supervised learning as well. In general, when W is close to block diagonal, it can be shown that data points are tightly clustered in the eigenspace spanned by the first few eigenvectors of Δ (Ng et al., 2001a; Meila & Shi, 2001), leading to various spectral clustering algorithms.

Perhaps the most interesting and substantial connection to the methods we propose here is the graph mincut approach proposed by Blum and Chawla (2001). The starting point for this work is also a weighted graph G, but the semi-supervised learning problem is cast as one of finding a minimum st-cut, where negative labeled data is connected (with large weight) to a special source node s, and positive labeled data is connected to a special sink node t. A minimum st-cut, which is not necessarily unique, minimizes the L^1 objective function $E_1(f) = \frac{1}{2}\sum_{i,j} w_{ij}\,|f(i) - f(j)|$ and corresponds to a function $f : V \to \{-1, +1\}$; the solutions can be obtained using linear programming. The corresponding random field model is a "traditional" field over the label space $\{-1, +1\}$, but the field is pinned on the labeled entries. Because of this constraint, approximation methods based on rapidly mixing Markov chains that apply to the ferromagnetic Ising model unfortunately cannot be used. Moreover, multi-label extensions are generally NP-hard in this framework. In contrast, the harmonic solution can be computed efficiently using matrix methods, even in the multi-label case, and inference for the Gaussian random field can be efficiently and accurately carried out using loopy belief propagation (Weiss et al., 2001).

4. Incorporating Class Prior Knowledge

To go from f to labels, the obvious decision rule is to assign label 1 to node i if $f(i) > \frac{1}{2}$, and label 0 otherwise. We call this rule the *harmonic threshold* (abbreviated "thresh" below). In terms of the random walk interpretation, if $f(i) > \frac{1}{2}$, then starting at i, the random walk is more likely to reach a positively labeled point before a negatively labeled point. This decision rule works well when the classes are well separated. However in real datasets, classes are often not ideally separated, and using f as is tends to produce severely unbalanced classification.

The problem stems from the fact that W, which specifies the data manifold, is often poorly estimated in practice and does not reflect the classification goal. In other words, we should not "fully trust" the graph structure. The class priors are a valuable piece of complementary information. Let's assume the desirable proportions for classes 1 and 0 are q and $1 - q$, respectively, where these values are either given by an "oracle" or estimated from labeled data. We adopt a simple procedure called *class mass normalization* (CMN) to adjust the class distributions to match the priors. Define the mass of class 1 to be $\sum_i f_u(i)$, and the mass of class 0 to be $\sum_i(1 - f_u(i))$. Class mass normalization scales these masses so that an unlabeled point i is classified as class 1 iff

$$q\frac{f_u(i)}{\sum_i f_u(i)} > (1-q)\frac{1 - f_u(i)}{\sum_i(1 - f_u(i))} \qquad (9)$$

This method extends naturally to the general multi-label case.

5. Incorporating External Classifiers

Often we have an external classifier at hand, which is constructed on labeled data alone. In this section we suggest how this can be combined with harmonic energy minimization. Assume the external classifier produces labels h_u on the unlabeled data; h_u can be 0/1 or soft labels in $[0, 1]$. We combine h_u with harmonic energy minimization by a simple modification of the graph. For each unlabeled node i in the original graph, we attach a "dongle" node which is a labeled node with value h_i, let the transition probability from i to its dongle be η, and discount all other transitions from i by $1 - \eta$. We then perform harmonic energy minimization on this augmented graph. Thus, the external classifier introduces "assignment costs" to the energy function, which play the role of vertex potentials in the random field. It is not difficult to show that the harmonic solution on the augmented graph is, in the random walk view,

$$f_u = (I - (1-\eta)P_{uu})^{-1}\left((1-\eta)P_{ul} f_l + \eta h_u\right) \qquad (10)$$

We note that throughout the paper we have assumed the labeled data to be noise free, and so clamping their values makes sense. If there is reason to doubt this assumption, it would be reasonable to attach dongles to labeled nodes as well, and to move the labels to these new nodes.

6. Learning the Weight Matrix W

Previously we assumed that the weight matrix W is given and fixed. In this section, we investigate *learning* weight functions of the form given by equation (1). We will learn the σ_d's from both labeled and unlabeled data; this will be shown to be useful as a feature selection mechanism which better aligns the graph structure with the data.

The usual parameter learning criterion is to maximize the likelihood of labeled data. However, the likelihood criterion is not appropriate in this case because the f values for labeled data are fixed during training, and moreover likelihood doesn't make sense for the unlabeled data because we do not have a generative model. We propose instead to use *average label entropy* as a heuristic criterion for parameter learning. The average label entropy $H(f)$ of the field f is defined as

$$H(f) = \frac{1}{u} \sum_{i=l+1}^{l+u} H_i(f(i)) \tag{11}$$

where $H_i(f(i)) = -f(i)\log f(i) - (1-f(i))\log(1-f(i))$ is the entropy of the field at the individual unlabeled data point i. Here we use the random walk interpretation of f, relying on the maximum principle of harmonic functions which guarantees that $0 < f(i) < 1$ for $i \geq l+1$. Small entropy implies that $f(i)$ is close to 0 or 1; this captures the intuition that a good W (equivalently, a good set of hyperparameters $\{\sigma_d\}$) should result in a *confident* labeling. There are of course many arbitrary labelings of the data that have low entropy, which might suggest that this criterion will not work. However, it is important to point out that we are constraining f on the labeled data—most of these arbitrary low entropy labelings are inconsistent with this constraint. In fact, we find that the space of low entropy labelings achievable by harmonic energy minimization is small and lends itself well to tuning the σ_d parameters.

There is a complication, however, which is that H has a minimum at 0 as $\sigma_d \to 0$. As the length scale approaches zero, the tail of the weight function (1) is increasingly sensitive to the distance. In the end, the label predicted for an unlabeled example is dominated by its nearest neighbor's label, which results in the following equivalent labeling procedure: (1) starting from the labeled data set, find the unlabeled point x_u that is closest to some labeled point x_l; (2) label x_u with x_l's label, put x_u in the labeled set and repeat. Since these are hard labels, the entropy is zero. This solution is desirable only when the classes are extremely well separated, and can be expected to be inferior otherwise.

This complication can be avoided by smoothing the transition matrix. Inspired by analysis of the PageRank algorithm in (Ng et al., 2001b), we replace P with the smoothed matrix $\tilde{P} = \varepsilon\mathcal{U} + (1-\varepsilon)P$, where $\mathcal{U}$ is the uniform matrix with entries $\mathcal{U}_{ij} = 1/(l+u)$.

We use gradient descent to find the hyperparameters σ_d that minimize H. The gradient is computed as

$$\frac{\partial H}{\partial \sigma_d} = \frac{1}{u} \sum_{i=l+1}^{l+u} \log\left(\frac{1-f(i)}{f(i)}\right) \frac{\partial f(i)}{\partial \sigma_d} \tag{12}$$

where the values $\partial f(i)/\partial\sigma_d$ can be read off the vector $\partial f_u/\partial\sigma_d$, which is given by

$$\frac{\partial f_u}{\partial \sigma_d} \quad = \quad (I - \tilde{P}_{uu})^{-1}\left(\frac{\partial \tilde{P}_{uu}}{\partial \sigma_d} f_u + \frac{\partial \tilde{P}_{ul}}{\partial \sigma_d} f_l\right) \tag{13}$$

using the fact that $dX^{-1} = -X^{-1}(dX)X^{-1}$. Both $\partial\tilde{P}_{uu}/\partial\sigma_d$ and $\partial\tilde{P}_{ul}/\partial\sigma_d$ are sub-matrices of $\partial\tilde{P}/\partial\sigma_d = (1-\varepsilon)\frac{\partial P}{\partial \sigma_d}$. Since the original transition matrix P is obtained by normalizing the weight matrix W, we have that

$$\frac{\partial p_{ij}}{\partial \sigma_d} \quad = \quad \frac{\frac{\partial w_{ij}}{\partial \sigma_d} - p_{ij}\sum_{n=1}^{l+u}\frac{\partial w_{in}}{\partial \sigma_d}}{\sum_{n=1}^{l+u} w_{in}} \tag{14}$$

Finally, $\frac{\partial w_{ij}}{\partial \sigma_d} = 2w_{ij}(x_{di} - x_{dj})^2/\sigma_d^3$.

In the above derivation we use f_u as label probabilities directly; that is, $p(\text{class}(x_i) = 1) = f_u(i)$. If we incorporate class prior information, or combine harmonic energy minimization with other classifiers, it makes sense to minimize entropy on the combined probabilities. For instance, if we incorporate a class prior using CMN, the probability is given by

$$\bar{f}(i) = \frac{q(u - \sum f_u)f_u(i)}{q(u - \sum f_u)f_u(i) + (1-q)\sum f_u(1 - f_u(j))} \tag{15}$$

and we use this probability in place of $f(i)$ in (11). The derivation of the gradient descent rule is a straightforward extension of the above analysis.

7. Experimental Results

We first evaluate harmonic energy minimization on a handwritten digits dataset, originally from the Cedar Buffalo binary digits database (Hull, 1994). The digits were preprocessed to reduce the size of each image down to a 16×16 grid by down-sampling and Gaussian smoothing, with pixel values ranging from 0 to 255 (Le Cun et al., 1990). Each image is thus represented by a 256-dimensional vector. We compute the weight matrix (1) with $\sigma_d = 380$. For each labeled set size l tested, we perform

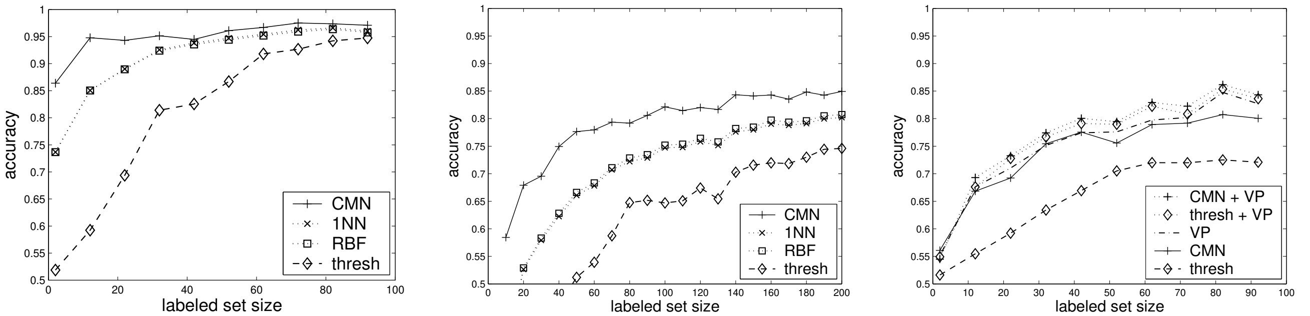

Figure 3. Harmonic energy minimization on digits "1" vs. "2" (left) and on all 10 digits (middle) and combining voted-perceptron with harmonic energy minimization on odd vs. even digits (right)

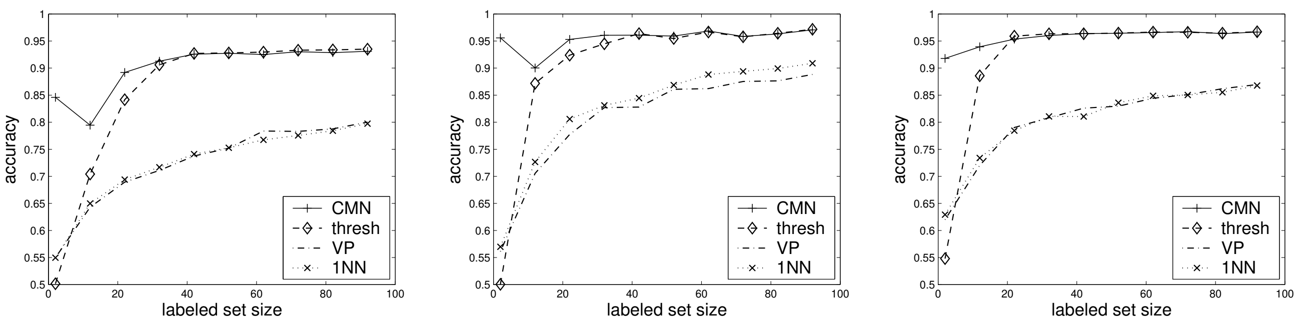

Figure 4. Harmonic energy minimization on PC vs. MAC (left), baseball vs. hockey (middle), and MS-Windows vs. MAC (right)

10 trials. In each trial we randomly sample labeled data from the entire dataset, and use the rest of the images as unlabeled data. If any class is absent from the sampled labeled set, we redo the sampling. For methods that incorporate class priors q, we estimate q from the labeled set with Laplace ("add one") smoothing.

We consider the binary problem of classifying digits "1" vs. "2," with 1100 images in each class. We report average accuracy of the following methods on unlabeled data: thresh, CMN, 1NN, and a radial basis function classifier (RBF) which classifies to class 1 iff $W_{ul} f_l > W_{ul}(1 - f_l)$. RBF and 1NN are used simply as baselines. The results are shown in Figure 3. Clearly thresh performs poorly, because the values of $f_u(j)$ are generally close to 1, so the majority of examples are classified as digit "1". This shows the inadequacy of the weight function (1) based on pixel-wise Euclidean distance. However the relative rankings of $f_u(j)$ are useful, and when coupled with class prior information significantly improved accuracy is obtained. The greatest improvement is achieved by the simple method CMN. We could also have adjusted the decision threshold on thresh's solution f_u, so that the class proportion fits the prior q. This method is inferior to CMN due to the error in estimating q, and it is not shown in the plot. These same observations are also true for the experiments we performed on several other binary digit classification problems.

We also consider the 10-way problem of classifying digits "0" through '9'. We report the results on a dataset with intentionally unbalanced class sizes, with 455, 213, 129, 100, 754, 970, 275, 585, 166, 353 examples per class, respectively (noting that the results on a balanced dataset are similar). We report the average accuracy of thresh, CMN, RBF, and 1NN. These methods can handle multi-way classification directly, or with slight modification in a one-against-all fashion. As the results in Figure 3 show, CMN again improves performance by incorporating class priors.

Next we report the results of document categorization experiments using the 20 newsgroups dataset. We pick three binary problems: PC (number of documents: 982) vs. MAC (961), MS-Windows (958) vs. MAC, and baseball (994) vs. hockey (999). Each document is minimally processed into a "tf.idf" vector, without applying header removal, frequency cutoff, stemming, or a stopword list. Two documents u, v are connected by an edge if u is among v's 10 nearest neighbors or if v is among u's 10 nearest neighbors, as measured by cosine similarity. We use the following weight function on the edges:

$$w_{uv} = \exp\left(-\frac{1}{0.03}\left(1 - \frac{u^\top v}{|u||v|}\right)\right) \qquad (16)$$

We use one-nearest neighbor and the voted perceptron algorithm (Freund & Schapire, 1999) (10 epochs with a linear kernel) as baselines–our results with support vector machines are comparable. The results are shown in Figure 4. As before, each point is the average of 10 random trials. For this data, harmonic energy minimization performs much better than the baselines. The improvement from the class prior, however, is less significant. An explanation for why this approach to semi-supervised learning is so effective on the newsgroups data may lie in the common use of quotations within a topic thread: document u_2 quotes part of document u_1, u_3 quotes part of u_2, and so on. Thus, although documents far apart in the thread may be quite different, they are linked by edges in the graphical representation of the data, and these links are exploited by the learning algorithm.

7.1. Incorporating External Classifiers

We use the voted-perceptron as our external classifier. For each random trial, we train a voted-perceptron on the labeled set, and apply it to the unlabeled set. We then use the 0/1 hard labels for dongle values h_u, and perform harmonic energy minimization with (10). We use $\eta = 0.1$.

We evaluate on the artificial but difficult binary problem of classifying odd digits vs. even digits; that is, we group “1,3,5,7,9” and “2,4,6,8,0” into two classes. There are 400 images per digit. We use second order polynomial kernel in the voted-perceptron, and train for 10 epochs. Figure 3 shows the results. The accuracy of the voted-perceptron on unlabeled data, averaged over trials, is marked VP in the plot. Independently, we run thresh and CMN. Next we combine thresh with the voted-perceptron, and the result is marked thresh+VP. Finally, we perform class mass normalization on the combined result and get CMN+VP. The combination results in higher accuracy than either method alone, suggesting there is complementary information used by each.

7.2. Learning the Weight Matrix W

To demonstrate the effects of estimating W, results on a toy dataset are shown in Figure 5. The upper grid is slightly tighter than the lower grid, and they are connected by a few data points. There are two labeled examples, marked with large symbols. We learn the optimal length scales for this dataset by minimizing entropy on unlabeled data.

To simplify the problem, we first tie the length scales in the two dimensions, so there is only a single parameter σ to learn. As noted earlier, without smoothing, the entropy approaches the minimum at 0 as $\sigma \to 0$. Under such conditions, the results of harmonic energy minimization are usually undesirable, and for this dataset the tighter grid “invades” the sparser one as shown in Figure 5(a). With smoothing, the “nuisance minimum” at 0 gradually disappears as the smoothing factor ε grows, as shown in Figure

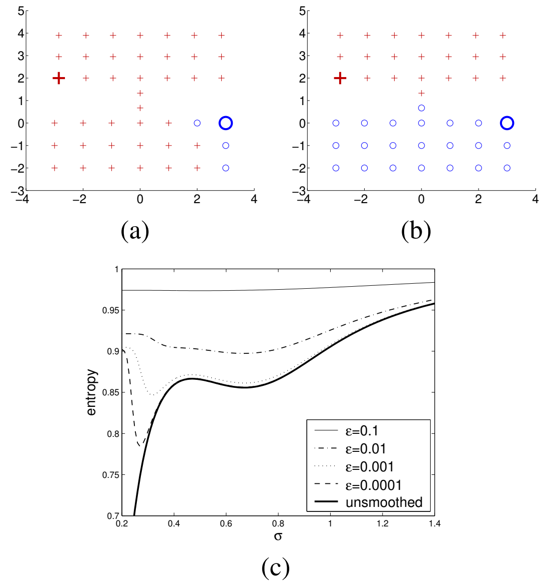

Figure 5. The effect of parameter σ on harmonic energy minimization. (a) If unsmoothed, $H \to 0$ as $\sigma \to 0$, and the algorithm performs poorly. (b) Result at optimal $\sigma = 0.67$, smoothed with $\varepsilon = 0.01$ (c) Smoothing helps to remove the entropy minimum.

5(c). When we set $\varepsilon = 0.01$, the minimum entropy is 0.898 bits at $\sigma = 0.67$. Harmonic energy minimization under this length scale is shown in Figure 5(b), which is able to distinguish the structure of the two grids.

If we allow a separate σ for each dimension, parameter learning is more dramatic. With the same smoothing of $\varepsilon = 0.01$, σ_x keeps growing towards infinity (we use $\sigma_x = 10^{16}$ for computation) while σ_y stabilizes at 0.65, and we reach a minimum entropy of 0.619 bits. In this case $\sigma_x \to \infty$ is legitimate; it means that the learning algorithm has identified the x-direction as irrelevant, based on both the labeled and unlabeled data. Harmonic energy minimization under these parameters gives the same classification as shown in Figure 5(b).

Next we learn σ's for all 256 dimensions on the “1” vs. “2” digits dataset. For this problem we minimize the entropy with CMN probabilities (15). We randomly pick a split of 92 labeled and 2108 unlabeled examples, and start with all dimensions sharing the same $\sigma = 380$ as in previous experiments. Then we compute the derivatives of σ for each dimension separately, and perform gradient descent to minimize the entropy. The result is shown in Table 1. As entropy decreases, the accuracy of CMN and thresh both increase. The learned σ's shown in the rightmost plot of Figure 6 range from 181 (black) to 465 (white). A small σ_i (black) indicates that the weight is more sensitive to variations in that dimension, while the opposite is true for large σ_i (white). We can discern the shapes of a black “1” and a white “2” in this figure; that is, the learned parameters

	H (bits)	CMN	thresh
start	0.6931	97.25 ± 0.73 %	94.70 ± 1.19 %
end	0.6542	98.56 ± 0.43 %	98.02 ± 0.39 %

Table 1. Entropy of CMN and accuracies before and after learning σ's on the "1" vs. "2" dataset.

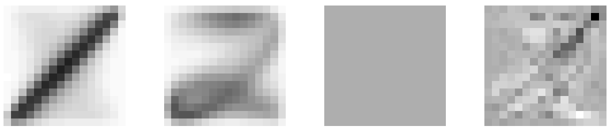

Figure 6. Learned σ's for "1" vs. "2" dataset. From left to right: average "1", average "2", initial σ's, learned σ's.

exaggerate variations within class "1" while suppressing variations within class "2". We have observed that with the default parameters, class "1" has much less variation than class "2"; thus, the learned parameters are, in effect, compensating for the relative tightness of the two classes in feature space.

8. Conclusion

We have introduced an approach to semi-supervised learning based on a Gaussian random field model defined with respect to a weighted graph representing labeled and unlabeled data. Promising experimental results have been presented for text and digit classification, demonstrating that the framework has the potential to effectively exploit the structure of unlabeled data to improve classification accuracy. The underlying random field gives a coherent probabilistic semantics to our approach, but this paper has concentrated on the use of only the mean of the field, which is characterized in terms of harmonic functions and spectral graph theory. The fully probabilistic framework is closely related to Gaussian process classification, and this connection suggests principled ways of incorporating class priors and learning hyperparameters; in particular, it is natural to apply evidence maximization or the generalization error bounds that have been studied for Gaussian processes (Seeger, 2002). Our work in this direction will be reported in a future publication.

References

Belkin, M., & Niyogi, P. (2002). Using manifold structure for partially labelled classification. *Advances in Neural Information Processing Systems, 15*.

Blum, A., & Chawla, S. (2001). Learning from labeled and unlabeled data using graph mincuts. *Proc. 18th International Conf. on Machine Learning*.

Boykov, Y., Veksler, O., & Zabih, R. (2001). Fast approximate energy minimization via graph cuts. *IEEE Trans. on Pattern Analysis and Machine Intelligence, 23*.

Chapelle, O., Weston, J., & Schölkopf, B. (2002). Cluster kernels for semi-supervised learning. *Advances in Neural Information Processing Systems, 15*.

Chung, F., & Yau, S. (2000). Discrete Green's functions. *Journal of Combinatorial Theory (A)* (pp. 191–214).

Doyle, P., & Snell, J. (1984). *Random walks and electric networks*. Mathematical Assoc. of America.

Freund, Y., & Schapire, R. E. (1999). Large margin classification using the perceptron algorithm. *Machine Learning, 37(3)*, 277–296.

Hull, J. J. (1994). A database for handwritten text recognition research. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 16*.

Kondor, R. I., & Lafferty, J. (2002). Diffusion kernels on graphs and other discrete input spaces. *Proc. 19th International Conf. on Machine Learning*.

Le Cun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Howard, W., & Jackel, L. D. (1990). Handwritten digit recognition with a back-propagation network. *Advances in Neural Information Processing Systems, 2*.

Meila, M., & Shi, J. (2001). A random walks view of spectral segmentation. *AISTATS*.

Ng, A., Jordan, M., & Weiss, Y. (2001a). On spectral clustering: Analysis and an algorithm. *Advances in Neural Information Processing Systems, 14*.

Ng, A. Y., Zheng, A. X., & Jordan, M. I. (2001b). Link analysis, eigenvectors and stability. *International Joint Conference on Artificial Intelligence (IJCAI)*.

Seeger, M. (2001). *Learning with labeled and unlabeled data* (Technical Report). University of Edinburgh.

Seeger, M. (2002). PAC-Bayesian generalization error bounds for Gaussian process classification. *Journal of Machine Learning Research, 3*, 233–269.

Shi, J., & Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 22*, 888–905.

Szummer, M., & Jaakkola, T. (2001). Partially labeled classification with Markov random walks. *Advances in Neural Information Processing Systems, 14*.

Weiss, Y., , & Freeman, W. T. (2001). Correctness of belief propagation in Gaussian graphical models of arbitrary topology. *Neural Computation, 13*, 2173–2200.

Yu, S. X., & Shi, J. (2001). Grouping with bias. *Advances in Neural Information Processing Systems, 14*.