Clustering and efficient use of unlabeled examples

Martin Szummer
MIT AI Lab & CBCL
Cambridge, MA 02139
szummer@ai.mit.edu

Tommi Jaakkola
MIT AI Lab
Cambridge, MA 02139
tommi@ai.mit.edu

Abstract

Efficient learning with partially labeled data involves extracting structure from large unlabeled set and combining this information with limited labeled examples. A typical albeit unstated assumption in this context associates separable clusters in the unlabeled set with unique but unknown labels. When this assumption is valid, labeled examples are needed only to the extent that they can facilitate labeling of the clusters. We capture and formalize this intuition in a conditional probability model where soft clusters serve to regularize the labeling of the unlabeled examples. Clustering is achieved by defining a Markov diffusion process (cf. Tishby and Slonim, NIPS 2000). The associated time scale of this process determines the effective size of the clusters and is chosen through a margin based criterion that guarantees unambiguous classification of examples. We relate the time scale to the mixing time of the Markov process and extend the basic idea by combining multiple time scales to maximize classification accuracy. We demonstrate the performance of the approach on both real and synthetic datasets.

1 Representation based on Markov diffusion

To achieve good learning performance, the data must be encoded in a suitable representation matched to the learning algorithm. Typically, we are provided data points in a space, and a distance metric that measures pairwise similarity between points. The provided distance metric is often quite accurate locally, as it is relatively easy to characterize small perturbations in the data. However, over larger distances, the given metric is frequently inadequate, and hurts the performance of the many learning algorithms that rely on global distances. Fortunately, in problems with many data points (with or without labels), we can use the locally accurate metric to construct an improved global distance measure that reflects the density of the data. For example, the data may lie on a submanifold of the space, revealed by the density, and we should measure distances along the manifold. Intuitively, the distances are smaller in directions of high density, and larger in low-density directions.

We define a Markov diffusion process based on the locally accurate metric. The local metric defines probabilities of transitioning between two nearby points in one timestep, and we construct the global distance as the probability of transitioning between two points in t timesteps. Thus, we consider all the paths of length t on this graph.

Formally, consider a set of points $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ that need not be labeled. Construct a graph whose nodes correspond to data points, and whose undirected edges correspond to one step transitions. We only allow such immediate transitions from a point to its neighbors. Specifically, for each point, connect it with an undirected edge to its K nearest neighbors. Points in high density centers can be neighbors of many points and end up with more than K edges. Self-transitions back to the point itself are also included. Let the probability of transitioning from a state i at time t to a neighbor k at time $t+1$ be

$$p_{ik} = P(k|i) = \frac{1}{Z_i} \exp(-\beta d(\mathbf{x}_i, \mathbf{x}_k)) \ \forall t,$$

where $d(\mathbf{x}_i, \mathbf{x}_k)$ is the local distance between the points, β is a positive real parameter, and Z_i normalizes probabilities so they sum to 1, namely $Z_i = \sum_k \exp(-\beta d(\mathbf{x}_i, \mathbf{x}_k))$.

Thus, we exponentiate distances to obtain probabilities. This relation reconciles the additive nature of distances with the multiplicative combination of probabilities when taking multiple steps.

Note that the transition probability p_{ik} and p_{ki} is not symmetric, because the normalization Z_i typically differs. As a rule of thumb, $p_{ik} > p_{ki}$ when $\mathbf{x}_k$ lies in a higher-density region than $\mathbf{x}_i$.

Denote the probability of transitioning from i at time 0 to k at time t with $p_{ik}^t = P(k^t|i^0) = P(k|i)$. The last notation omits the t and 0 superscripts from k and i respectively. If we organize all transition probabilities as a matrix $\mathbf{A}$ whose i,k-th entry is p_{ik}, we can simply use a matrix power to calculate

$$p_{ik}^t = [\mathbf{A}^t]_{ik}.$$

The matrix is row stochastic so that rows sum to 1. We represent a point in terms of its probabilities of having transitioned from each of the other points. The k-th point is represented in terms of the probabilities of originating at any point i, namely $P(i^0|k) \propto P(k^t|i^0)P(i^0) \propto P(k^t|i^0)$, since we assume uniform starting probabilites. We can use the compact vector notation $[p_{1|k}, \ldots, p_{N|k}] = \frac{1}{Z_k'}[p_{1k}^t, \ldots, p_{Nk}^t]$, where Z_k' normalizes the sum of components to 1. Two points are close if these origination probabilities are similar.

To fully specify the representation, we must choose K, β, and t. The first two parameters influence the one-step probabilities. The t parameter regulates the amount of smoothing due to diffusion. As t increases, the probabilities p_{ik}^t approach the stationary distribution, which is independent of i. We defer details til section 3.

2 Parameter estimation for classification

We now assume that we are given a partially labeled data set $\{(\mathbf{x}_1, \tilde{y}_1), \ldots, (\mathbf{x}_L, \tilde{y}_L), \mathbf{x}_{L+1}, \ldots, \mathbf{x}_N\}$ and we wish to classify the unlabeled points. Typically, the number of labeled points L is much smaller than the total points N. We want to employ our representation and introduce a parameter $p_{y|i} = P(y|\mathbf{x}_i^0)$ for each component. The parameters $p_{y|i}$ are probabilities in $[0,1]$. The classifier has the form

$$P(y|\mathbf{x}_k^t) = \sum_i P(y|\mathbf{x}_i^0) P(\mathbf{x}_i^0|\mathbf{x}_k^t).$$

For brevity, we will henceforth drop the timestep superscripts. We discuss three techniques for choosing them: maximum likelihood with EM, maximizing average margin subject to constraints, and maximum entropy discriminiation.

2.1 EM estimation

We maximize the conditional log-likelihood

$$\sum_{l=1}^{L} \log P(\tilde{y}_l|\mathbf{x}_l) = \sum_{l=1}^{L} \log \sum_{i=1}^{N} P(\tilde{y}_l|i)P(i|\mathbf{x}_l) \tag{1}$$

where the first summation is only over the labeled examples. Since $P(i|\mathbf{x}_l)$ are fixed, this objective function is jointly concave in the free parameters and lends itself to a unique maximum value. The concavity also guarantees that this optimization is easily performed via the EM algorithm [1].

Let $p_{i|l}$ be the soft assignment for component i given $(\mathbf{x}_l, \tilde{y}_l)$, i.e., $p_{i|l} = P(i|\mathbf{x}_l, \tilde{y}_l) \propto P(\tilde{y}_l|i)P(i|\mathbf{x}_l)$. The EM algorithm iterates between the E-step, where $p_{i|l}$ are recomputed from the current estimates of $P(y|i)$, and the M-step where we update $P(y|i) \leftarrow \sum_{l:\tilde{y}_l=y} p_{i|l} / \sum_l p_{i|l}$.

The runtime of this algorithm is $\mathcal{O}(L\,N)$. The discriminative formulation suggests that EM will provide reasonable parameter estimates $P(y|i)$ for classification purposes. The quality of the solution, as well as the potential for overfitting, is contingent on the smoothness of the representation, specifically the origination probabilities $P(\mathbf{x}_i|\mathbf{x}_k)$. Note, however, that whether or not $P(y|i)$ will converge to the extreme values 0 or 1 is not an indication of overfitting. Actual classification decisions for unlabeled examples $\mathbf{x}_i$ (included in the expansion) need to be made on the basis of $P(y|\mathbf{x}_i)$ and not on the basis of $P(y|i)$, which function as parameters.

2.2 Margin based estimation

An alternative discriminative formulation is also possible, one that is more sensitive to the decision boundary rather than probability values associated with the labels. To this end, consider the conditional probability $P(y|\mathbf{x}_k) = \sum_i P(y|i)P(i|\mathbf{x}_k)$. The decisions are made on the basis of the sign of the discriminant function

$$f(\mathbf{x}_k) = P(y=1|\mathbf{x}_k) - P(y=-1|\mathbf{x}_k) = \sum_{i=1}^{N} w_i P(i|\mathbf{x}_k) \tag{2}$$

where $w_i = P(y=1|i) - P(y=-1|i)$. This is similar to a linear classifier and there are many ways of estimating the weights w_i discriminatively. The weights should remain bounded, however, i.e., $w_i \in [-1, 1]$, so long as we wish to maintain the probabilistic interpretation of the parameters. Estimation algorithms with Euclidean norm regularization such as SVMs would not be appropriate in this sense.

For separable problems, we propose a simple linear program that maximizes the margin γ for labeled points, which is the smallest distance between the decision boundary and the point. The maximization is subject to classifying the labeled points correctly:

$$\begin{array}{rcll} \max\limits_{w_i} \gamma & & \text{subject to} & \\ \tilde{y}_l f(\mathbf{x}_l) & \geq & \gamma & \forall l \in [1 \ldots L] \\ w_i & \leq & 1 & \forall i \in [1 \ldots N] \\ -w_i & \leq & 1 & \end{array}$$

Solutions of linear programs are achieved at extremal points of the set. The Kuhn-Karush-Tucker conditions require that the optimal w_i will equal 1 or -1 except for points that satisfy the margin constraint with equality. Thus the majority of weights labels will be hard.

Problems with very few labeled examples are typically separable, especially for moderate values of t, when the representation is not overly smooth. If the problem is nonseparable, the margin and weights will be 0 and this formulation is not useful. One possibility is to introduce a individual margin variable γ_i for each point and optimize the average margin. The margins are bounded to have magnitude less than 1, reducing the risk that any single point would dominate the average margin. Individual margins are equivalent to adding linear slack variables and optimizing a common margin as above. However, if a common margin is desired together with slack variables, maximum entropy discrimination provides a framework to do so [5, 6], and we recommend this latter technique in the non-separable case.

2.3 Sample size requirements

Here we quantify the sample size that is needed for accurate estimation of the labels for the unlabeled examples. Since we are considering a transduction problem, i.e., finding labels for already observed examples, the sample size requirements can be assessed directly in terms of the diffusion matrix. As before, the probabilities $P(i|k)$ and $P(i|j)$ are diffusion probabilities starting in i ending in k and j respectively.

Lemma 1 *Let $d_{jk} = \sum_{i=1}^{n} |P(i|j) - P(i|k)|$. The $V(\gamma)$ dimension of the transductive classifier is upper bounded by the number of connected components of a graph with n nodes and adjacency matrix A, where $A_{jk} = 1$ if $d_{jk} \leq \gamma$ and zero otherwise.*

Proof: The discriminant function $f(\mathbf{x}_j)$ in the two-class case is given by

$$f(\mathbf{x}_j) = \sum_{i=1}^{n} P(i|j)[q(y=1|j) - q(y=-1|j)] \tag{3}$$

Assume that $y_j f(\mathbf{x}_j) \geq \gamma$ for all j. We wish to evaluate the number of complete labelings $\{y_j\}$ consistent with these margin constraints.

We establish first that all examples $\mathbf{x}_j$ and $\mathbf{x}_k$ for which $d_{jk} \leq \gamma$ must have the same label. This follows directly from

$$\begin{aligned} |f(\mathbf{x}_j) - f(\mathbf{x}_k)| &\leq \sum_{i=1}^{n} |P(i|j) - P(i|k)| \, |q(y=1|j) - q(y=-1|j)| \\ &\leq \sum_{i=1}^{n} |P(i|j) - P(i|k)| = d_{jk} \end{aligned}$$

as this difference must be larger than γ for the discriminant functions to have different signs. Since any pair of examples for which $d_{jk} \leq \gamma$ share the same label, different labels can be assigned only to examples not connected by the $d_{jk} \leq \gamma$ relation.□

Given a dataset, and a desired classification margin γ we calculate the representation and let $r = V(\gamma)$ dimension. With high probability we can correctly classify the unlabeled points given $O(r \log r)$ labeled examples [3]. This can also be helpful to determine timescale t since it is reflected in the $V(\gamma)$, for example $V(\gamma) = N$ for $t = 0$ and $V(\gamma) = 1$ for $t = \infty$ for the full range of $\gamma \in [0, 2]$.

2.4 Examples

Consider an example (figure 1) of classification with Markov diffusion. We are given 2 labeled and 148 unlabeled points in an intertwining swiss-roll pattern. We set $K = 6$

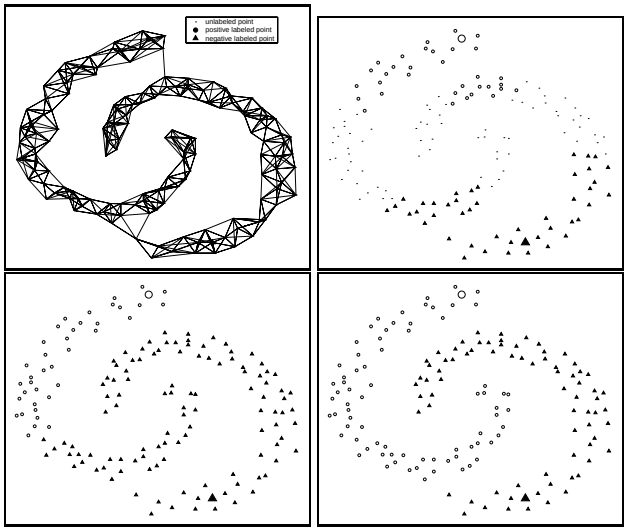

Figure 1: Topmost is the connectivity structure for symmetric 6-nearest neighbors. Below are dlassifications using Markov diffusion for t=3, 20, and 500 (top to bottom), estimated with EM. There are two labeled points (large circle, triangle) and 148 unlabeled points, most of which have been classified (small circles, triangles) but for small values of t the Markov diffusion may not have reached them, leaving them assigned 50/50 to both classes (small dots).

and $\beta = 10$ (the box has extent 2×2), and show three different timescales. At $t = 3$ the diffusion has not connected all points, so that some unlabeled points have no path to any labeled point. In the EM algorithm, their parameters do not affect the labeled data likelihood, and they then remain assigned equally to both classes. The other points have a path to only one of the classes, and are therefore fully assigned to that class. At $t = 20$ all points have paths to labeled points, however, the Markov process has not mixed well. Some paths may not follow the curved high-density structure, and instead cross between the two clusters. The triangle class dominates the assignment. When the Markov process is well-mixed at $t = 500$, the points are labeled as expected, even though labels changed back and forth for different ts. The parameter assignments are hard, but the class posteriors are weighted averages of these and soft.

3 Parameter choices for K, β and t

For sufficiently large K the graph will have no distinct connected components and if, in addition, all the distances corresponding to the edges in the graph are finite, the Markov process defined on the graph will be ergodic. In practice, the choice K seems to have little effect, e.g., on the resulting classification performance. This can be in part due to the fact that adjusting β can counter the effect from increasing the number of neighbors (e.g., for large β neighbors further away appear disconnected).

The smoothness of the diffusion representation also depends on the number of diffusion time steps t. This is a regularization parameter akin to the kernel width of a density estimator. In the limiting case $t = 1$, we employ only the initial neighborhood graph. In this case, the above classifier would reduce to a distance weighted K-nearest neighbor for points that have labeled neighbors; points without labeled neighbors would have uniform probabilities over the labels. If we also increase K to include all the points we obtain the

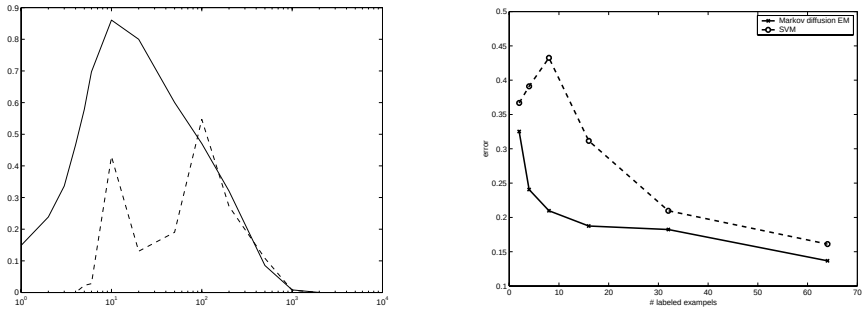

Figure 2: Left: Average margin (solid) and minimum margin (dashed, multiplied by 10). Right: Test errors for 2–64 labeled examples for markov diffusion (bottom) and best SVM (top)

mixture distance or kernel expansion representation [6].

In the limiting case $t = \infty$ the representation for each node converges to a uniform weighting over the points in the same connected component. Put another way, if point $\mathbf{x}_i$ and $\mathbf{x}_k$ belong to the same connected component of size N_{C_k} then $P(\mathbf{x}_i^0|\mathbf{x}_k^\infty) = 1/N_{C_k}$, otherwise the probability is 0. The likelihood is maximized for all assignments where the parameters $P(y|i)$ average to the class priors within their respective connected components.

We can make more appropriate choices for t with a few unsupervised heuristics. If $t+1$ equals the diameter of a singly connected graph, then we ensure that $P(\mathbf{x}_i^0|\mathbf{x}_k^t) > 0$ so each point influences every other point. However, this scheme ignores transition probabilities. Instead, the *mixing time* of a graph measures the time it takes to approach the stationary distribution p_k^∞. The graph mixes faster the smaller the second largest eigenvalue λ_2 of the transition matrix $\mathbf{A}$ (the largest eigenvalue is always 1). To reach within ϵ in half L1 distance from the stationary distribution, we must have [8]

$$t \geq \max_i \frac{1}{1-\lambda_2}(\ln \frac{1}{p_i^\infty} + \ln \frac{1}{\epsilon})$$

where p_i^∞ is the stationary probability at node i. Similarly to [4] we wish to choose t so that we are relatively far from the overall stationary distribution. The rate of mutual information dissipation used by [4] to identify cluster development does not, however, suffice as a recipe for choosing the overall time scale t.

Good choices of t for classification are not independent of labels. For example, if labels change quickly over small distances, we want a sharper representation and a smaller t. Cross-validation could provide a supervised choice of t but requires many labeled points for accuracy, which we do not have here. It is also computationally expensive. Instead, we propose to choose t that maximizes the log likelihood (eq. 1), which is also equivalent to maximizing the average log(1 plus margin). We average this margin measure over *all* the points, both labeled and unlabeled to ensure a solid recovery of the labels most of the observed points.

Figure 2 shows both the average and minimum margins as a function of t, for the swissroll example. The average margin has a single peak, but occurs for smaller values of t than our subjectively preferred segmentation, which occurs for t closer to the highest peak of the minimum margin.

3.1 Multiple time scales

So far, we have employed a single global value of t. However, the desired smoothness may be different at different locations (akin to adaptive kernel widths [2]). At the simplest, if the graph has multiple connected components, we can set individual t for each component. Ideally, each point has its own time scale, and the choice of time scale is optimized jointly with the classifier parameters. Here we propose a restricted version of this criterion where we find individual time scales t_k for each unlabeled point but estimate the remaining parameters separately.

The principle by which we select the time scales for the unlabeled points encourages the node identities to become the only common correlates for the labels. More precisely, define $P(y|k, t_k)$ for any unlabeled point k as

$$P(y|k, t_k) = \frac{1}{Z_k} \sum_{i:\bar{y}_i=y} P(i|k, t_k) \tag{4}$$

where $Z_k = \sum_i P(i|k, t_k)$ and both summations are over *only* the labeled points. Moreover, let $P(y)$ be the overall probability over the labels across the unlabeled points or

$$P(y) = \sum_k P(k)P(y|k, t_k) \tag{5}$$

where $P(k)$ is the invariant stationary distribution over the nodes in the graph. Note that $P(y)$ remains a function of all the individual time scales for the unlabeled points. With these definitions, the principle for setting the time scales reduces to maximizing the mutual information between the label and the node identity:

$$\{t_1, \ldots, t_m\} = \arg\max_{t_1,\ldots,t_m} I(y; k) \tag{6}$$

$$= \arg\max_{t_1,\ldots,t_m} \left\{ H(y) - \sum_k P(k)H(y|k) \right\} \tag{7}$$

where $H(y)$ and $H(y|k)$ are the marginal and conditional entropies over the labels and are computed on the basis of $P(y)$ and $P(y|k, t_k)$, respectively. Note that the ideal setting of the time scales would be one that determines the labels for the unlabeled points uniquely on the basis of only the labeled examples while at the same time preserving the overall variability of the labels across the nodes. This would happen, for example, if the labeled examples fall on distinct connected components. The criterion can be optimized using an axis parallel search, where only discrete values of t_k need to be tried.

4 Experimental results

We applied the markov diffusion approach to partially labeled text classification, with few labeled documents but many unlabeled ones. Text documents are represented by high-dimensional vectors but only occupy low-dimensional manifolds, so we expect markov diffusion to be beneficial. We used the `mac` and `windows` subsets from the 20 newsgroups dataset[1]. There were 958 and 961 examples in the two classes, with 7511 dimensions after rare words were removed. We estimated the manifold dimensionality at 8. Consequently $K = 10$ seemed a suitable choice of neighborhood size, and also led to a graph with a single connected component. The histogram of distances to the 10 nearest neighbor is peaked at 1.3, so we choose $\beta = 0.6$ for a reasonable falloff. We plotted the decay of

[1] Processed as 20news-18827, `http://www.ai.mit.edu/~jrennie/20Newsgroups/`, removing rare words, duplicate documents, and performing tf-idf mapping.

mutual information as a function of t and chose $t = 8$, after verifying histograms of the entropy of the representation vectors. We trained both the EM and the linear programming formulation, using 2–64 labeled points, treating all remaining points as unlabeled. We trained on 10 random splits balanced for class labels, and tested on a fixed separate set of 987 points. Results in figure 2 show a clear advantage over the best SVM out of linear and Gaussian SVMs for different kernel widths and values of C. The linear programming training runs slightly faster than EM but produces slightly worse test errors.

References

[1] Miller D., Uyar T. (1996) A Mixture of Experts Classifer with Learning Based on Both Labelled and Unlabelled Data. NIPS 9, pp. 571–577.

[2] Scott, David (1992) Multivariate density estimation : theory, practice, and visualization. Wiley.

[3] Avrim Blum, Shuchi Chawla. (2001) Learning from Labeled and Unlabeled Data using Graph Mincuts. ICML.

[4] Tishby N; Slonim N. (2000) Data clustering by Markovian relaxation and the Information Bottleneck Method. NIPS 13

[5] Jaakkola T., Meila M., and Jebara T. (1999) Maximum entropy discrimination. NIPS 12.

[6] Szummer, M; Jaakkola, T. (2000) Kernel expansions with unlabeled examples. NIPS 13.

[7] Tenenbaum, J, de Silva, V; Langford J. (2000) A Global Geometric Framework for Nonlinear Dimensionality Reduction. Science 290 (5500): 2319-2323.

[8] Chung, Fan. (1997) Spectral graph theory.