The Sample Complexity of Semi-Supervised Learning with Nonparametric Mixture Models

Chen Dan, Liu Leqi, Bryon Aragam, Pradeep Ravikumar, Eric Xing
Carnegie Mellon University

Carnegie Mellon University

Overview

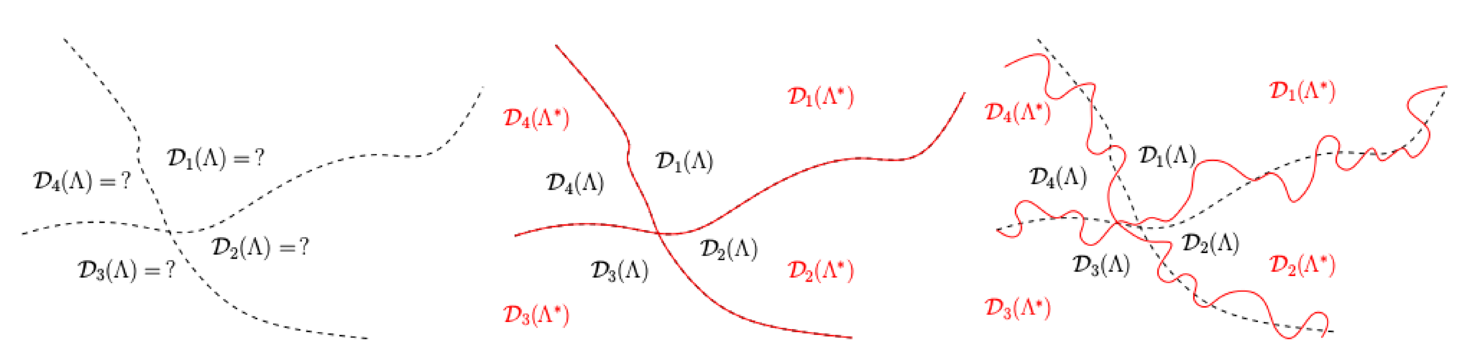
- A novel framework for analyzing the sample complexity of semi-supervised learning (SSL) in general, nonparametric settings.
- Establish $\Omega(K \log K)$ sample complexity for learning the class assignment and provide conditions under which the resulting classifier converges to the Bayes classifier.
- Provide efficient algorithms in learning the class assignment and illustrate their performance on real and simulated data.

SSL as Permutation Learning

Let K be the number of classes in the output space $\mathcal{Y}$. We formulate SSL as follows (see Figure below):

1. Use the unlabeled data to learn a K-component nonparametric mixture model Λ that approximates the unlabeled data density F^*;
2. Use the labeled data to learn an assignment $\pi : [K] \to \mathcal{Y}$ between decision regions $\mathcal{D}_b(\Lambda)$ and classes α_k;
3. Given a test point X, first assign to a decision region, then use π to assign a label.

The pair (Λ, π) thus defines a classifier $g_{\Lambda,\pi} : \mathcal{X} \to \mathcal{Y}$ that we analyze.



Assumptions

We make the following general assumptions:

1. **Nonparametric.** The class conditional distributions $\mathbb{P}(X \mid Y)$ can be anything, and we do not assume any parametric form.
2. **Multi-class ($K > 2$).** Existing techniques based on Neyman-Pearson classification no longer apply.
3. **Unknown $\mathbb{P}(X)$, $\mathbb{P}(X \mid Y)$.** Both the unlabeled data distribution and class conditionals are unknown and unidentified.

These assumptions generalize existing work, which either assume $K = 2$ or that the unlabeled data distribution is either known, or approximately known.

References

Castelli, V. and Cover, T. M. (1996), IEEE Trans. Inform. Theory 42(6): 2102-2117.
Aragam, B., Dan, C., Ravikumar, P., and Xing, E.P. (2018), arXiv: 1802.04397.
Rigollet, P. (2007), JMLR 8(Jul): 1369-1392.
Singh, A., Nowak, R. and Zhu. X. (2009), NeurIPS: 1513-1520.

Sample Complexity

Maximum likelihood. The maximum likelihood estimator (MLE) is given by:

$$\widehat{\pi}_{\text{MLE}} \in \operatorname*{argmax}_{\pi} \ell_n(\pi; \Lambda), \quad \ell_n(\pi; \Lambda) := \frac{1}{n}\sum_{i=1}^{n} \log \lambda_{\pi(Y)} f_{\pi(Y)}(X).$$

Given Λ, the notation $\mathbb{E}_*\ell(\pi; \Lambda, X, Y) = \mathbb{E}_* \log \lambda_{\pi(Y)} f_{\pi(Y)}(X)$ denotes the expectation of the *misspecified* log-likelihood with respect to the *true* distribution. Define the "gap"

$$\Delta_{\text{MLE}}(\Lambda) := \mathbb{E}_*\ell(\pi^*; \Lambda, X, Y) - \max_{\pi \neq \pi^*} \mathbb{E}_*\ell(\pi; \Lambda, X, Y). \tag{1}$$

For any function $a : \mathbb{R} \to \mathbb{R}$, define the usual Fenchel-Legendre dual $a^*(t) = \sup_{s\in\mathbb{R}}(st - a(s))$. Let $U_b = \log \lambda_b f_b(X)$ and $\beta_b(s) = \log \mathbb{E}_* \exp(sU_b)$.

Theorem 1 (Sample complexity of MLE). *Suppose that $\lambda_k^* = 1/K$ for each k, $\Delta_{\text{MLE}} > 0$, and*

$$n \geq K \log(K/\delta)\Big[1 + \frac{4}{\inf_b \beta_b^*(\Delta_{\text{MLE}}/3)}\Big].$$

Then $\mathbb{P}(\widehat{\pi}_{\text{MLE}} = \pi^*) \geq 1 - \delta$.

Majority vote. The majority vote estimator (MV) is given by a simple majority vote over amongst the labels in each decision region. For any Λ, define $m_b := |i : X^{(i)} \in \mathcal{D}_b(\Lambda)|$ and $\chi_{bj}(\Lambda) := \frac{1}{m_b}\sum_{i=1}^{n} 1(Y^{(i)} = j, X^{(i)} \in \mathcal{D}_b(\Lambda))$, where $1(\cdot)$ is the indicator function. Similar to the MLE, our results for MV depend crucially on a "gap" quantity, given by

$$\Delta_{\text{MV}}(\Lambda) := \inf_b \Big\{\mathbb{E}_*\chi_{bb}(\Lambda) - \max_{j\neq b} \mathbb{E}_*\chi_{bj}(\Lambda)\Big\}. \tag{2}$$

Theorem 2 (Sample complexity of MV). *Suppose that $\mathbb{P}(X \in \mathcal{D}_b(\Lambda)) = 1/K$ for each k, $\Delta_{\text{MV}} > 0$, and*

$$n \geq K \log(K/\delta)\Big[1 + \frac{18}{\Delta_{\text{MV}}^2}\Big].$$

Then $\mathbb{P}(\widehat{\pi}_{\text{MV}} = \pi^*) \geq 1 - \delta$.

Classification Error

We can further bound the classification error of the classifier in terms of the Wasserstein distance $W_1(\Lambda, \Lambda^*)$ between Λ and Λ^* as follows:

Theorem 3 (Classification error). *Let $g^* = g_{\Lambda^*,\pi^*}$ denote the Bayes classifier. If $\pi^*(\alpha_b) = \operatorname{argmin}_i d_{\text{TV}}(f_i, f_b^*)$ then there is a constant $C > 0$ depending on K and Λ^* such that*

$$\mathbb{P}(g_{\Lambda,\pi^*}(X) \neq Y) \leq \mathbb{P}(g^*(X) \neq Y) + C \cdot W_1(\Lambda, \Lambda^*) + \sum_b |\lambda_{\pi^*(\alpha_b)} - \lambda_b^*|.$$

This theorem allows for the possibility that the mixture model Λ learned from the unlabeled data is not the same as Λ^*. It is thus necessary to assume that the mismatch between Λ and Λ^* is not so bad that the closest density f_i to f_b^* is something other than $f_{\pi^*(\alpha_b)}$.

Algorithms

Define $C_k = \{i : Y^{(i)} = \alpha_k\}$.

MLE. The MLE can be found via the Hungarian algorithm, by exploiting a connection with max weight perfect matching in the bipartite graph $G = (V_{K,K}, w)$ with $w(k, k') = \sum_{i\in C_k} \log\big(\lambda_{k'} f_{k'}(X^{(i)})\big)$.

Majority vote. This is straightforward to compute.

Greedy. Assign the kth class to $\widehat{\pi}_{\text{G}}(\alpha_k) = \operatorname{argmax}_{k'\in[K]} w(k, k') = \operatorname{argmax}_{k'\in[K]} \sum_{i\in C_k} \log\big(\lambda_{k'} f_{k'}(X^{(i)})\big)$. This greedy heuristic can be viewed as a "soft interpolation" of $\widehat{\pi}_{\text{MLE}}$ and $\widehat{\pi}_{\text{MV}}$.

Performance of the Algorithms

To test the performance of the three algorithms, we consider three settings: (i) Mixtures of Gaussians, (ii) A nonparametric mixture model, and (iii) MNIST. $\mathbb{P}(\hat{\pi} = \pi^*)$ is evaluated in two settings: (i) $\Lambda = \Lambda^*$ and (ii) $\Lambda \neq \Lambda^*$.

In terms of classification accuracy, each algorithm was compared with a canonical supervised baseline for MNIST, the LeNet convolutional neural network. As shown below, all three estimators attain higher accuracy with fewer labeled samples, but the accuracy plateaus around 95% since $\Lambda \neq \Lambda^*$.

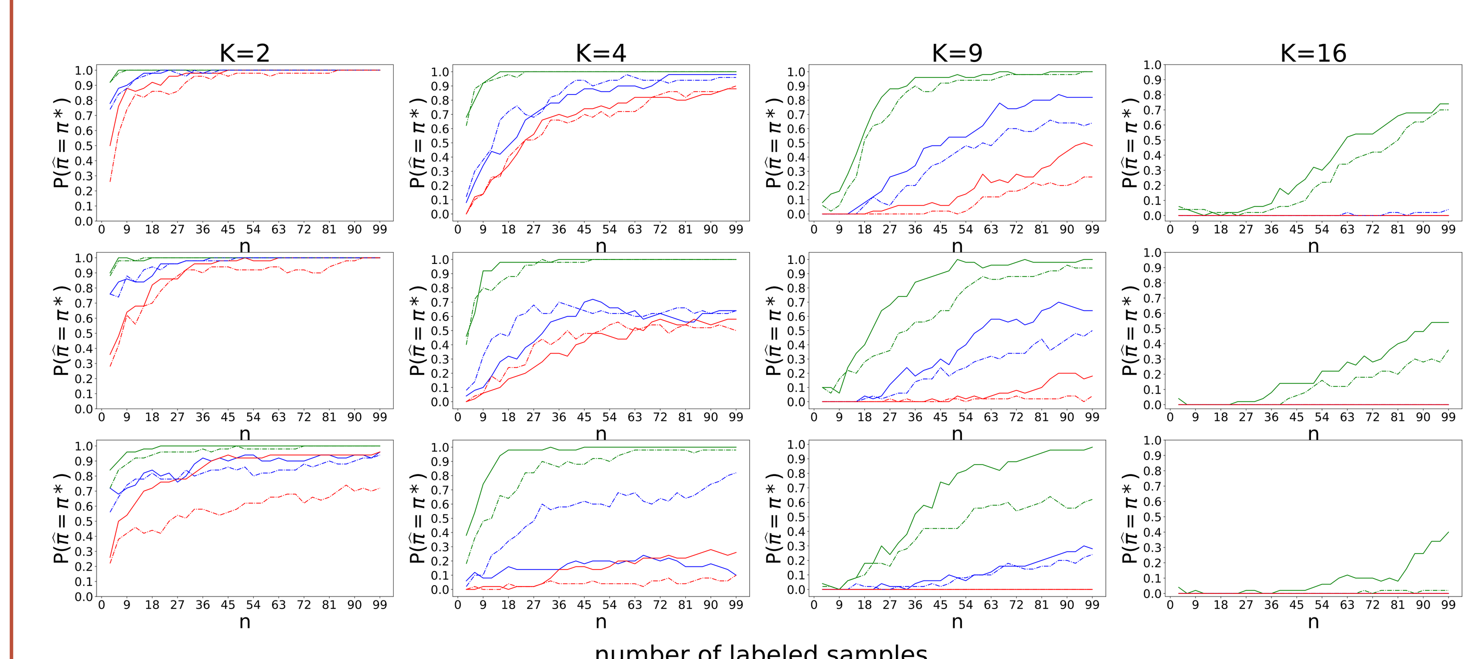


Figure 1

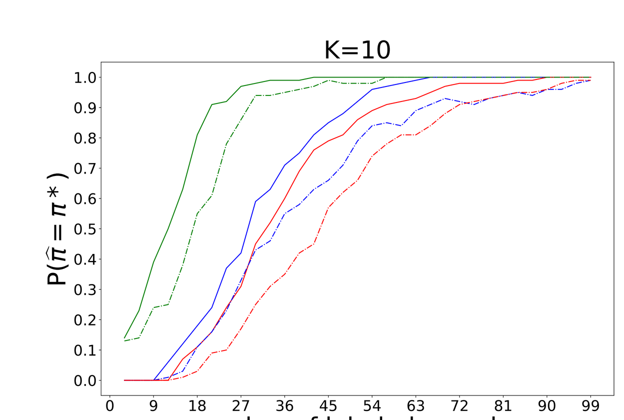


Figure 2

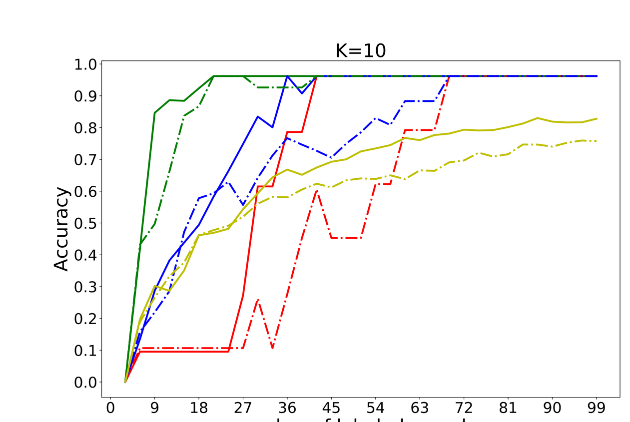


Figure 3

Figure 1 (Mixture of Gaussian) and Figure 2 (MNIST) report performance of MLE (Hungarian - Green; Greedy - Blue) and MV (Red) on learning the class assignment. Solid line and dashed line correspond to the performance when $\Lambda^* = \Lambda$ and $\Lambda^* \neq \Lambda$, respectively. In Figure 1, columns correspond to the number of classes K; rows correspond to decreasing separation; e.g. the bottom rows in each figure are the least separated. Figure 3 shows classification accuracy of the SSL estimators and LeNet (Yellow) on MNIST.