Iterative Hybrid Algorithm for Semi-supervised Classification

Martin SAVESKI
Supervised by professor Thierry Artières

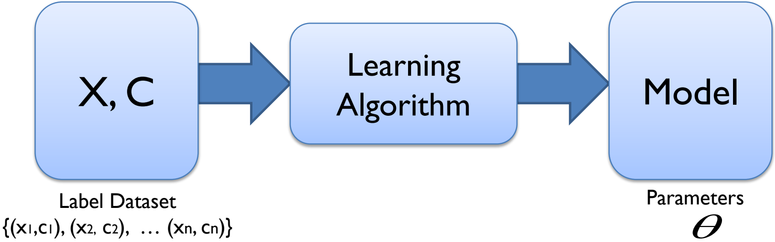
University Pierre and Marie Curie

June 19, 2012

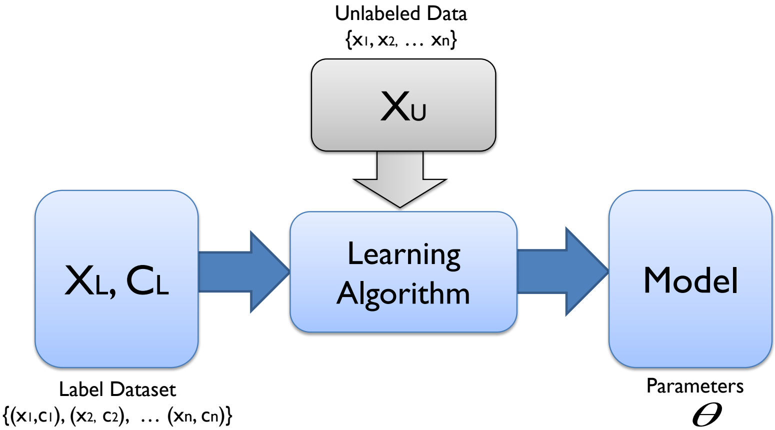
Outline

- Intro to Semi-supervised Learning
- The Iterative Hybrid Algorithm
- Other methods
- Experiments
- Performance comparison and observations

Classical Supervised Learning Scenario



Semi-Supervised Learning



How to use the *unlabeled* data to build better classifiers?

Generative v.s. Discriminative Models

Generative Models

- Model how samples from a particular class are generated
- p modeling inputs, hidden variables, and outputs jointly
- Strong modeling power, *can easily handle missing values*

$$L_G(\theta) = p(X, C, \theta) = p(\theta) \prod_{n=1}^{N} p(x_n, c_n | \theta_{c_n}).$$

Generative v.s. Discriminative Models

Generative Models

- Model how samples from a particular class are generated
- p modeling inputs, hidden variables, and outputs jointly
- Strong modeling power, *can easily handle missing values*

$$L_G(\theta) = p(X, C, \theta) = p(\theta) \prod_{n=1}^{N} p(x_n, c_n | \theta_{c_n}).$$

Discriminative Models

- Concerned with defining the boundaries between the classes
- Directly optimize the boundary
- Tend to achieve better accuracy

$$L_D(\theta) = p(C | X, \theta) = \prod_{n=1}^{N} p(c_n | x_n, \theta).$$

Generative v.s. Discriminative Models

Generative Models

- Model how samples from a particular class are generated
- p modeling inputs, hidden variables, and outputs jointly
- Strong modeling power, *can easily handle missing values*

$$L_G(\theta) = p(X, C, \theta) = p(\theta) \prod_{n=1}^{N} p(x_n, c_n|\theta_{c_n}).$$

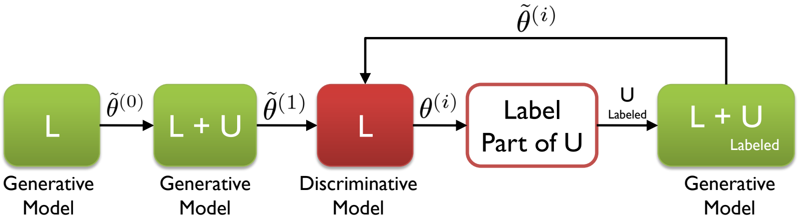
Discriminative Models

- Concerned with defining the boundaries between the classes
- Directly optimize the boundary
- Tend to achieve better accuracy

$$L_D(\theta) = p(C|X, \theta) = \prod_{n=1}^{N} p(c_n|x_n, \theta).$$

No easy way to combine them!

Input: Labeled and Unlabeled data



Iterative Hybrid Algorithm (more formally)

1. Learn $\tilde{\theta}$ on $L \rightarrow \tilde{\theta}^{(0)}$, by maximizing the following objective function:

$$\sum_{x \in L} \log p(x|c, \tilde{\theta})$$

2. Learn $\tilde{\theta}$ on $L \cup U \rightarrow \tilde{\theta}^{(1)}$, starting from $\tilde{\theta}^{(0)}$, maximizing:

$$\sum_{x \in L} \log p(x|c, \tilde{\theta}) + \lambda \sum_{x \in U} \log \sum_{c'} p(x|c', \tilde{\theta})$$

Iterative Hybrid Algorithm (more formally)

Loop n number of iterations, or until convergence:

1. Learn θ on $L \rightarrow \theta^{(i)}$, starting from $\tilde{\theta}^{(i)}$, maximizing:

$$-\frac{1}{2}||\theta - \tilde{\theta}^{(i)}||^2 + \sum_{x \in L} \log p(c|x, \theta)$$

2. Use $\theta^{(i)}$ to label part of $U \rightarrow U_{Labeled}$, where the labels are assigned as:

$$x \rightarrow c = \arg\max_{c} p(c|x, \theta^{(i)})$$

3. Learn $\tilde{\theta}$ on $L + U_{Labeled} \rightarrow \tilde{\theta}^{(i)}$, maximizing:

$$\sum_{x \in L} \log p(x|c, \tilde{\theta}) + \lambda \sum_{x \in U_{Labeled}} \log p(x|c, \tilde{\theta})$$

Other methods

Hybrid Model (Bishop and Lasserre, 2007)

- Multi-criteria objective function
- Combines generative and discriminative models with specific priors
- Optimizes:

$$p(\theta, \tilde{\theta}) \prod_{n \in L} p(C_n | X_n, \theta) \prod_{m \in L \cup U} p(X_m | \tilde{\theta})$$

Other methods

Hybrid Model (Bishop and Lasserre, 2007)

- Multi-criteria objective function
- Combines generative and discriminative models with specific priors
- Optimizes:

$$p(\theta, \tilde{\theta}) \prod_{n \in L} p(C_n | X_n, \theta) \prod_{m \in L \cup U} p(X_m | \tilde{\theta})$$

Entropy Minimization (Grandvalet and Bengio, 2005)

- Uses the label entropy on unlabeled data as a regularizer.
- Assumes a prior which prefers minimal class overlap
- Optimizes:

$$\sum_{x \in L} \log p(c|x, \theta) + \lambda \sum_{x \in U} \sum_{c' \in C} p(c'|x, \theta) \log p(c'|x, \theta)$$

Experiments

Data Set

- Synthetic Data (2 dimensions, 2 classes)
- Generated by elongated Gaussian distributions
- 2 labeled points per class
- 200 unlabeled per class
- 200 test samples per class

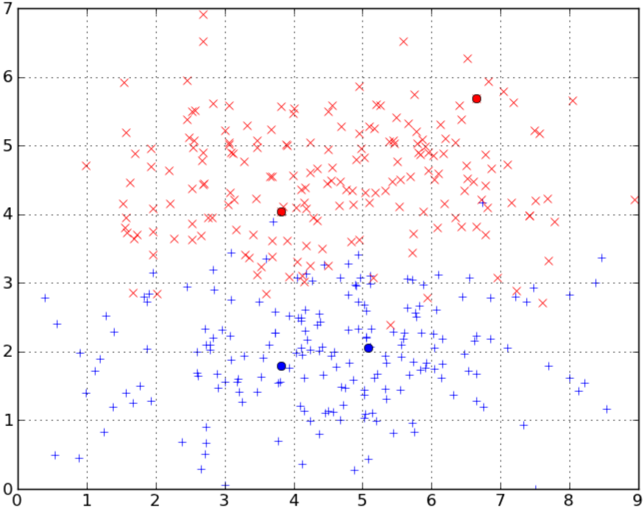
Model

- $p(x|c) \rightarrow$ Iso-tropic Gaussian distribution
- Symmetric distribution (model misspecification)

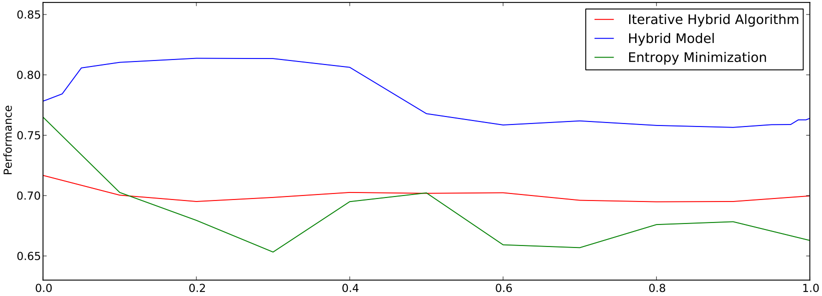
Setup

- Generate random data and label random points
- Run all algorithms for all hyper-parameter values

Example Data Set



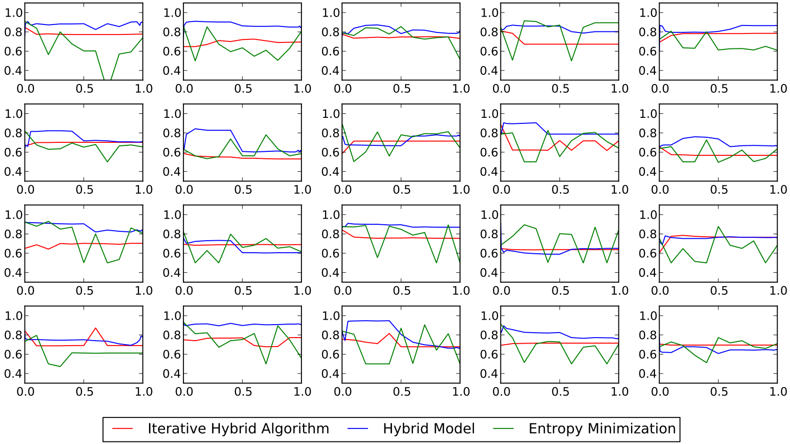
Results with Two Labeled Points



Parameters have different semantics, not directly comparable

Hybrid Model > Iterative Hybrid Algorithm > Entropy Minimization

Results with Two Labeled Points (cont.)



Hard to fix the hyper-parameters

Unstable behavior of the Entropy Minimization method

IHA and HM have stable behavior (iterative process possible)

Particular Cases

- Manually fixed points
- Boundary induced by the labeled points far from the real one
- Important feature
  Overlap on the x axis between labeled points
  - If NO Overlap $\rightarrow$ both perform well
  - If Overlap $\rightarrow$ Hybrid Model superior

Particular Cases (HM superior scenario)

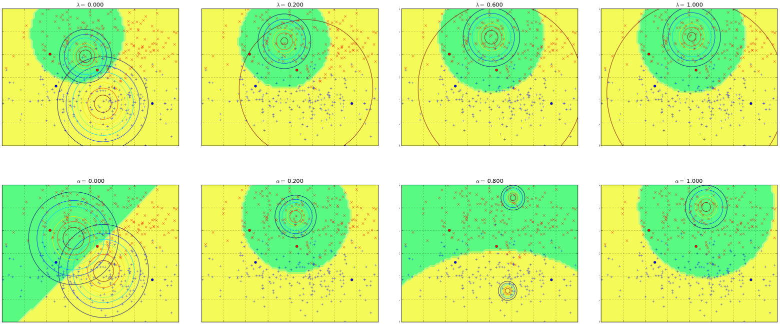
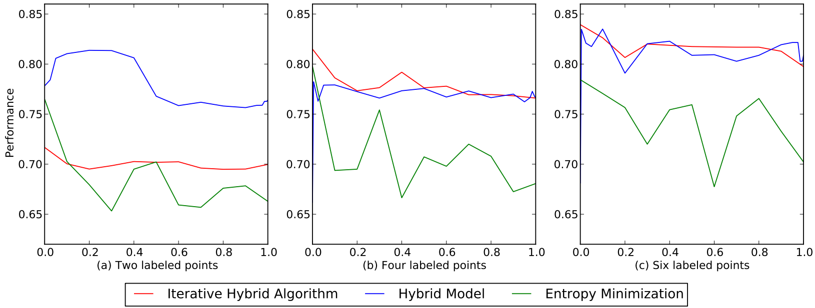


Figure 5: A case where there is an overlap overlap between the labeled points of each class on the x axis. The Iterative Hybrid Algorithm is shown on the top and the Hybrid Model on the bottom. The Iterative Hybrid Algorithm correctly classifies the labeled points, but fails to converge to the real boundary between the classes. However, the Hybrid Model for $\alpha = 0.8$ converges to a satisfactory solution.

Top: Iterative Hybrid Algorithm
Bottom: Hybrid Model

Increasing the number of labeled examples



As the number of labeled examples increases

- Difference between IHA and HM diminishes
- Entropy Minimization, improved performance, but still behind

To sum up

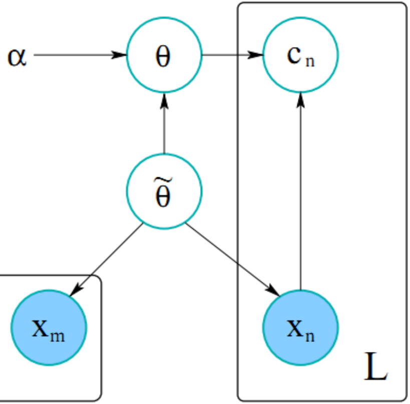
- Iterative Algorithm for combining generative and discriminative models
- Compared with two other methods (HM and EM)
- Experiments on synthetic data
- IHA dominates Entropy Minimization, but outperformed by the Hybrid Model
- Difference vanishes as $|L|$ increases

It is your turn now ...

Questions?

Hybrid Model (details)

$$
\begin{aligned}
q(\mathbf{X}, \mathbf{C}, \boldsymbol{\theta}, \widetilde{\boldsymbol{\theta}}) &= q(\mathbf{X}_{\mathrm{L}}, \mathbf{C}_{\mathrm{L}}, \mathbf{X}_{\mathrm{U}}, \boldsymbol{\theta}, \widetilde{\boldsymbol{\theta}}) \\
&= p(\boldsymbol{\theta}, \widetilde{\boldsymbol{\theta}})\, p(\mathbf{C}_{\mathrm{L}} | \mathbf{X}_{\mathrm{L}}, \boldsymbol{\theta})\, p(\mathbf{X}_{\mathrm{L}}, \mathbf{X}_{\mathrm{U}} | \widetilde{\boldsymbol{\theta}}) \\
&= p(\boldsymbol{\theta}, \widetilde{\boldsymbol{\theta}}) \prod_{n \in \mathrm{L}} p(\mathbf{c}_n | \mathbf{x}_n, \boldsymbol{\theta}) \prod_{m \in \mathrm{L} \cup \mathrm{U}} p(\mathbf{x}_m | \widetilde{\boldsymbol{\theta}})
\end{aligned}
$$



Entropy Minimization

Entropy Minimization (Grandvalet and Bengio, 2005)

- Uses the label entropy on unlabeled data as a regularizer.
- Assumes a prior which prefers minimal class overlap
- Optimizes:

$$\sum_{x \in L} \log p(c|x,\theta) + \lambda \sum_{x \in U} \sum_{c' \in C} p(c'|x,\theta) \log p(c'|x,\theta)$$

- Using U to estimate the **conditional Entropy** $H(Y|X)$ (measure of class overlap)

Why Discriminative?

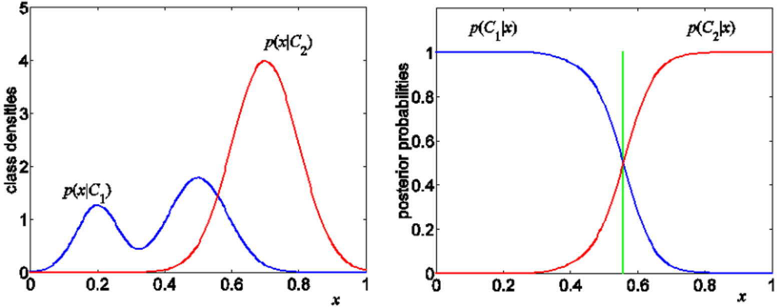


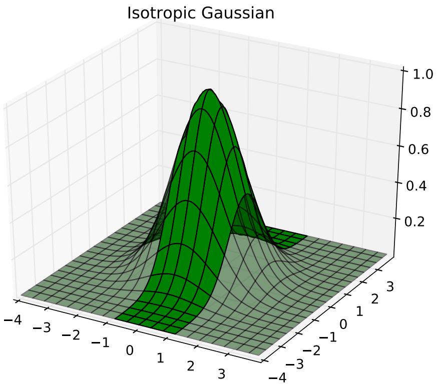
Figure 1.5: **Generative model vs discriminative model.** Taken from [7]. Example of the class-conditional densities for 2 classes having a single input variable x (left plot) together with the corresponding posterior probabilities (right plot). Note that the left-hand mode of the class conditional density $p(x|C_1)$ shown in blue on the left plot, has no effect on the posterior probabilities. The vertical green line in the right plot shows the decision boundary in x that gives the minimum misclassification rate.

Conditional Learning

$$p(\mathbf{c}|\mathbf{x},\boldsymbol{\theta}) = \frac{p(\mathbf{x},\mathbf{c}|\boldsymbol{\theta}_\mathbf{c})}{p(\mathbf{x}|\boldsymbol{\theta})} = \frac{p(\mathbf{x},\mathbf{c}|\boldsymbol{\theta}_\mathbf{c})}{\sum_{\mathbf{c}'} p(\mathbf{x},\mathbf{c}'|\boldsymbol{\theta}_{\mathbf{c}'})}$$

$$L_C(\boldsymbol{\theta}) = p(\boldsymbol{\theta}) \prod_{n=1}^{N} p(\mathbf{c}_n|\mathbf{x}_n,\boldsymbol{\theta}) = p(\boldsymbol{\theta}) \prod_{n=1}^{N} \frac{p(\mathbf{x}_n,\mathbf{c}_n|\boldsymbol{\theta}_{\mathbf{c}_n})}{\sum_{\mathbf{c}} p(\mathbf{x}_n,\mathbf{c}|\boldsymbol{\theta}_\mathbf{c})}$$

Iso-tropic Gaussian



$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{\frac{-((x-\mu_x)^2)+(y-\mu_y)^2)}{2\sigma^2}}$$