A Taxonomy of Semi-Supervised Learning Algorithms

Olivier Chapelle

Max Planck Institute for Biological Cybernetics

December 2005

MAX-PLANCK-GESELLSCHAFT

BIOLOGISCHE KYBERNETIK

Outline

1. Introduction
2. Generative models
3. Low density separation
4. Graph based methods
5. Unsupervised learning
6. Conclusions

Outline

1. Introduction
2. Generative models
3. Low density separation
4. Graph based methods
5. Unsupervised learning
6. Conclusions

The semi-supervised learning (SSL) paradigm

We consider here the problem of binary classification.

Definition (Supervised learning)

Given a training set $\{(\mathbf{x}_i, y_i)\}$ estimate a decision function (or more generally a probability $P(y|\mathbf{x})$).

Definition (Semi-supervised learning)

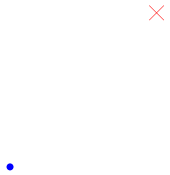
Same goal as supervised learning, but in addition a set of unlabeled points $\{\mathbf{x}'_i\}$ is available.

Typically, much more unlabeled data than labeled data.

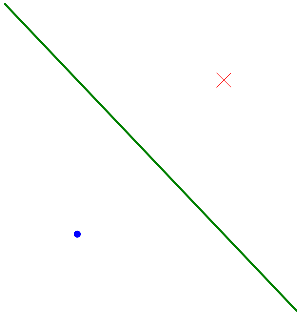
Note: differs from the related notion of transduction.

Are unlabeled data useful ?

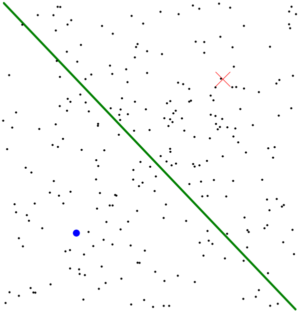
Are unlabeled data useful ?



Are unlabeled data useful ?

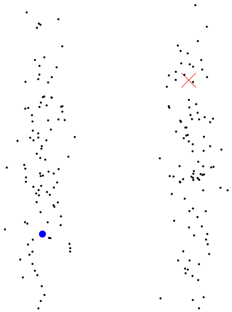


Are unlabeled data useful ?

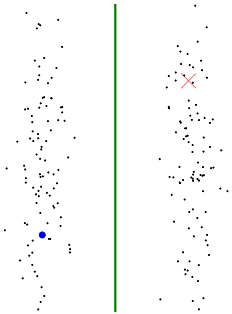


No

Are unlabeled data useful ?

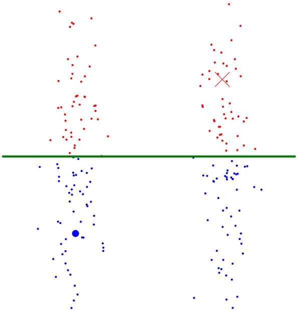


Are unlabeled data useful ?



Yes !

Are unlabeled data useful ?



Well, not sure.

The cluster assumption

Need for assumption

Standard supervised assumption

Two points which are near are likely to be of the same label.

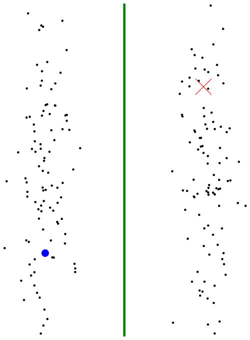
Cluster assumption

Two points which are in the same cluster (i.e. which are linked by a high density path) are likely to be of the same label.

Equivalently,

Low density separation assumption

The decision boundary should lie in a low density region.



The cluster assumption

- This assumption seems sensible for a lot of real world datasets.
- It is used in nearly all SSL algorithms, but most of the time implicitly.
- No equivalent formulation for regression.
  It seems that SSL is not very useful for regression.

Infinite amount of unlabeled data

A core fundamental question that an SSL algorithm should tackle is

What should I do if I knew exactly the marginal distribution $P(\mathbf{x})$?

Semi-supervised algorithms should be seen as a special case of this limiting case.

Unfortunately, lack of research in this direction.
Probably due to historical reasons: for supervised learning, when $P(\mathbf{x}, y)$ is known, classification is trivial.

Infinite amount of unlabeled data

A core fundamental question that an SSL algorithm should tackle is

What should I do if I knew exactly the marginal distribution $P(\mathbf{x})$?

Semi-supervised algorithms should be seen as a special case of this limiting case.

Unfortunately, lack of research in this direction.
Probably due to historical reasons: for supervised learning, when $P(\mathbf{x}, y)$ is known, classification is trivial.

Generative vs discriminative learning

Generative learning

1. For each y, learn the class conditional density $P(\mathbf{x}|y,\theta)$ (and also the class prior $P(y|\theta)$).
2. For a test point $\mathbf{x}$, compute $P(y|\mathbf{x},\theta) \propto P(\mathbf{x}|y,\theta)P(y|\theta)$. [Bayes rule]

Discriminative learning

Learn directly $P(y|\mathbf{x})$ (or a decision function).

- Generative learning was popular in the 70s.
- Main advantage of discriminative learning: it avoids the difficult step of modeling class conditional densities.
- Nowadays, discriminative classifiers are usually preferred.

Generative vs discriminative learning

Generative learning

1. For each y, learn the class conditional density $P(\mathbf{x}|y, \theta)$ (and also the class prior $P(y|\theta)$).
2. For a test point $\mathbf{x}$, compute $P(y|\mathbf{x}, \theta) \propto P(\mathbf{x}|y, \theta)P(y|\theta)$. [Bayes rule]

Discriminative learning

Learn directly $P(y|\mathbf{x})$ (or a decision function).

- Generative learning was popular in the 70s.
- Main advantage of discriminative learning: it avoids the difficult step of modeling class conditional densities.
- Nowadays, discriminative classifiers are usually preferred.

Outline

1 Introduction

2 Generative models

3 Low density separation

4 Graph based methods

5 Unsupervised learning

6 Conclusions

Generative models

It is straightforward to use unlabeled data in a generative model:

Find the model parameters θ maximizing the log-likelihood of the labeled and unlabeled data,

$$\sum_i \log(\underbrace{P(\mathbf{x}_i|y_i,\theta)P(y_i|\theta)}_{P(\mathbf{x}_i,y_i|\theta)}) + \sum_i \log(\underbrace{\sum_y P(\mathbf{x}'_i|y,\theta)P(y|\theta)}_{P(\mathbf{x}'_i|\theta)}).$$

Simplest example: each class has a Gaussian distribution.

This is a missing value problem.
$\longrightarrow$ Can be learned with the Expectation-Maximization (EM) algorithm.

Generative models

It is straightforward to use unlabeled data in a generative model:

Find the model parameters θ maximizing the log-likelihood of the labeled and unlabeled data,

$$\sum_i \log(\underbrace{P(\mathbf{x}_i|y_i,\theta)P(y_i|\theta)}_{P(\mathbf{x}_i,y_i|\theta)}) + \sum_i \log(\underbrace{\sum_{\color{red}y} P(\mathbf{x}'_i|{\color{red}y},\theta)P({\color{red}y}|\theta)}_{P(\mathbf{x}'_i|\theta)}).$$

Simplest example: each class has a Gaussian distribution.

This is a missing value problem.
$\longrightarrow$ Can be learned with the Expectation-Maximization (EM) algorithm.

Generative learning - EM

EM is used to maximize the likelihood of model with hidden variables.

EM algorithm for SSL

- E-step: compute $q_i(y) = P(y|\mathbf{x}'_i, \theta)$
- M-step: maximize over θ,

$$\sum_i \log(P(\mathbf{x}_i|y_i, \theta)P(y_i|\theta)) + \sum_i \sum_y q_i(y) \log(P(\mathbf{x}'_i|y, \theta)P(y|\theta))$$

Nice interpretation and relation to self-learning:

- E-step: estimate the labels according to the current decision function.
- M-step: estimate the decision function with the current labels.

Toy example

Class conditional density is Gaussian.

Demo EM

Experiments on text classification

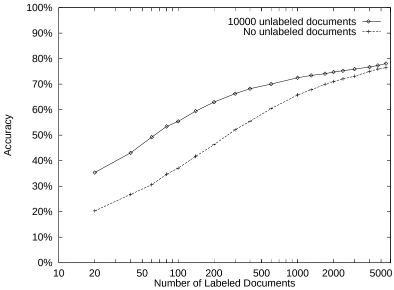
Nigam et al, *Text Classification from Labeled and Unlabeled Documents Using EM*, Machine Learning, 2000

- Bag of words representation
- Multinomial distribution

$$P(\mathbf{x}|, y, \theta) = \prod_{words} \theta_{w|y}^{x_w}$$

$\longrightarrow$ Naive Bayes classifier

- Several components per class
- 20 `Newsgroups` dataset



Intuition: SSL detects words co-occurrence.

Analysis of generative methods

Advantages

- Easy to use
- Unlabeled data are very useful.
  $\longrightarrow$ In the limit, they determine the decision boundary (labeled points are only useful for the direction).

Drawback

Usually, the model is misspecified.
$\longrightarrow$ There is no θ such that $P(\mathbf{x}) \equiv P(\mathbf{x}|\theta)$.
Unlabeled data can be misleading since Maximum Likelihood tries to model $P(\mathbf{x})$ rather than $P(y|\mathbf{x})$.

Note: the cluster assumption is not explicitly stated, but implied by standard models such as mixture of Gaussians.

Outline

1. Introduction
2. Generative models
3. Low density separation
4. Graph based methods
5. Unsupervised learning
6. Conclusions

Low density separation

Find a decision boundary which lies in low density regions (do not cut clusters).

For instance, find f with no training error and which minimizes

$$\max_{\mathbf{x},\ f(\mathbf{x})=0} P(\mathbf{x})$$

P is unknown in practice, but a kernel density estimate can be used.
$\longrightarrow$ Push the decision boundary away from the unlabeled points.

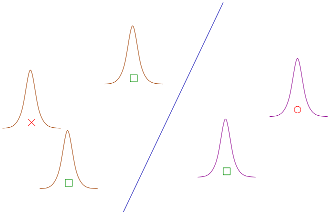
Low density separation

Find a decision boundary which lies in low density regions (do not cut clusters).

For instance, find f with no training error and which minimizes

$$\max_{\mathbf{x},\ f(\mathbf{x})=0} P(\mathbf{x})$$

P is unknown in practice, but a kernel density estimate can be used.
$\longrightarrow$ Push the decision boundary away from the unlabeled points.

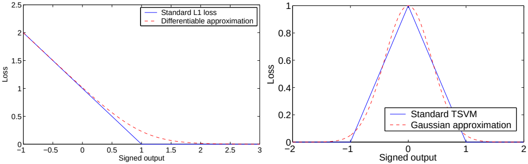


Transductive Support Vector Machines

Transductive Support Vector Machines (TSVM)

Maximize the margin on both labeled and unlabeled points:

$$\min_{\mathbf{w},b} \quad \overbrace{\underbrace{\mathbf{w}^2}_{\text{regularizer}} \quad + C \sum \underbrace{L(y_i(\mathbf{w}\cdot\mathbf{x}_i + b))}_{\text{labeled loss}}}^{\text{standard SVM}} + C' \sum \underbrace{L'(\mathbf{w}\cdot\mathbf{x}'_i + b)}_{\text{unlabeled loss}}$$



Main difficulty

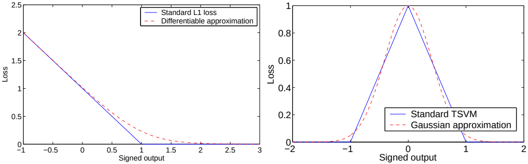
Non convex optimization problem $\longrightarrow$ local minima

Transductive Support Vector Machines

Transductive Support Vector Machines (TSVM)

Maximize the margin on both labeled and unlabeled points:

$$\min_{\mathbf{w},b} \overbrace{\underbrace{\mathbf{w}^2}_{\text{regularizer}} + C\sum \underbrace{L(y_i(\mathbf{w}\cdot\mathbf{x}_i + b))}_{\text{labeled loss}}}^{\text{standard SVM}} + C'\sum \underbrace{L'(\mathbf{w}\cdot\mathbf{x}'_i + b)}_{\text{unlabeled loss}}$$



Main difficulty

Non convex optimization problem $\longrightarrow$ local minima

Experiments

1. Toy problem, varying C'

Demo TSVM

2. Text classification
   - 10 most frequent categories of the Reuters dataset.
   - 17 labeled documents, 3299 unlabeled ones.
   - The average precsion/recall breakeven point went from 48.4% (SVM) to 60.8% (TSVM).

T. Joachims, *Transductive Inference for Text Classification using Support Vector Machines*, ICML 1999

Experiments

1. Toy problem, varying C'

Demo TSVM

2. Text classification
   - 10 most frequent categories of the Reuters dataset.
   - 17 labeled documents, 3299 unlabeled ones.
   - The average precsion/recall breakeven point went from 48.4% (SVM) to 60.8% (TSVM).

T. Joachims, *Transductive Inference for Text Classification using Support Vector Machines*, ICML 1999

Outline

1 Introduction

2 Generative models

3 Low density separation

4 Graph based methods

5 Unsupervised learning

6 Conclusions

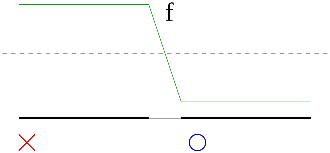
Measure based regularization

Finding a low density separation is a difficult problem.
$\longrightarrow$ Another approach to enforce the cluster assumption is to consider regularizers such as

$$\int ||\nabla f(\mathbf{x})|| P(\mathbf{x}) d\mathbf{x}$$

By doing so, the function

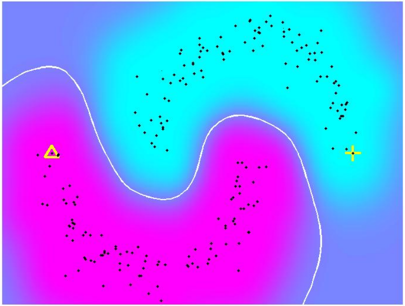
- does not change a lot in high density regions,
- is allowed to vary in low density regions.



Measure based regularization

Toy problem: "two moons"

- RBF network, centers = unlabeled points
- Kernel density estimate

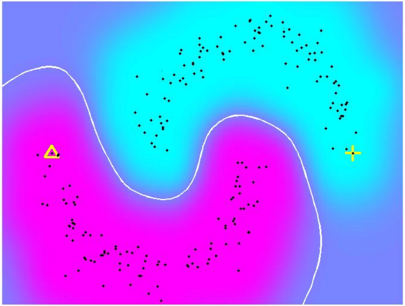


Smooth in high density $\not\Rightarrow$ decision boundary does not cut clusters.

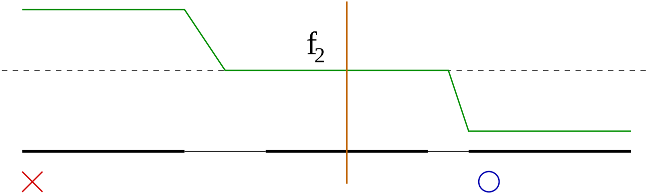
Measure based regularization

Toy problem: "two moons"

- RBF network, centers = unlabeled points
- Kernel density estimate



Smooth in high density $\not\Rightarrow$ decision boundary does not cut clusters.



Graph based approaches

Graph regularization

Construct a graph whose vertices are the labeled and unlabeled points, typically a (weighted) nearest neighbor graph and minimize

$$\sum_{i,j} W_{ij}(f(\mathbf{x}_i) - f(\mathbf{x}_j))^2 \qquad [W \text{ is the adjacency matrix}]$$

- Discretized version of the measure based regularization
- When f takes only binary values $\longrightarrow$ "cut" of the graph.

A lot of related algorithms based on different motivations

- Regularization [Belkin '02, Smola '03]
- Clustering
  - Graph min-cut [Blum '01, Joachims '03, Bach '03]
  - Spectral Clustering [Ng '01, Chapelle '02]
- Diffusion [Szummer '01, Zhu '02, Kondor '02, Zhou '03]

Graph based approaches

Graph regularization

Construct a graph whose vertices are the labeled and unlabeled points, typically a (weighted) nearest neighbor graph and minimize

$$\sum_{i,j} W_{ij}(f(\mathbf{x}_i) - f(\mathbf{x}_j))^2 \qquad [W \text{ is the adjacency matrix}]$$

- Discretized version of the measure based regularization
- When f takes only binary values $\longrightarrow$ "cut" of the graph.

A lot of related algorithms based on different motivations

- Regularization [Belkin '02, Smola '03]
- Clustering
  - Graph min-cut [Blum '01, Joachims '03, Bach '03]
  - Spectral Clustering [Ng '01, Chapelle '02]
- Diffusion [Szummer '01, Zhu '02, Kondor '02, Zhou '03]

Graph based approaches

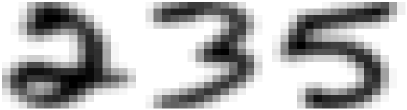
Works very well if the data lie on a low dimensional manifold.

Main difficulties

- Construction of the graph
- Gives a transductive solution (defined on the unlabeled points) and not an inductive one (defined everywhere).

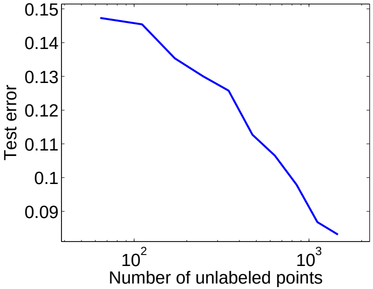
Handwritten digit recognition

- Handwritten digits (USPS)
- 256 dimensions
- Class 0 to 4 against 5 to 9
- 2007 samples



Low dimensional manifold (translations, rotations, ...)

50 labeled points, varying the number of unlabeled points.

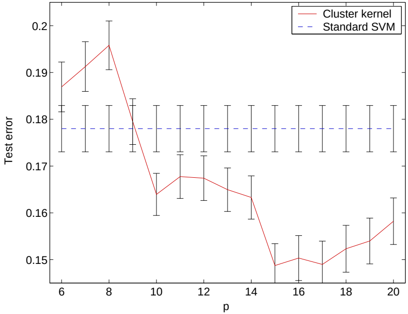


Handwritten digit recognition

O. Chapelle et al., *Cluster kernels for semi-supervised learning*, NIPS 2002

Kernel function for semi-supervised learning based on spectral clustering.

- Hyperparameter $p \approx$ corresponding to the number of clusters.
- Local minimum for $p = 10$, i.e. number of digits.



Outline

1. Introduction
2. Generative models
3. Low density separation
4. Graph based methods
5. Unsupervised learning
6. Conclusions

Unsupervised learning as a first step

Two steps procedure

1. Unsupervised learning (ignoring the labels)
   $\longrightarrow$ New distance / representation.
2. Supervised learning with the new distance / representation (ignoring the unlabeled points).

- Advantage: simple procedure using existing algorithms.
- Drawback: could be suboptimal.

A lot of possibilities: (spectral) clustering, change of distances, dimensionality reduction (PCA, LSI or non-linear).

Unsupervised learning as a first step

Two steps procedure

1. Unsupervised learning (ignoring the labels)
   $\longrightarrow$ New distance / representation.
2. Supervised learning with the new distance / representation (ignoring the unlabeled points).

- Advantage: simple procedure using existing algorithms.
- Drawback: could be suboptimal.

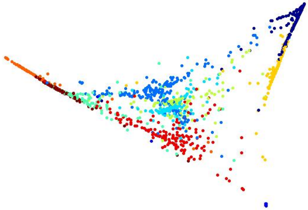
A lot of possibilities: (spectral) clustering, change of distances, dimensionality reduction (PCA, LSI or non-linear).

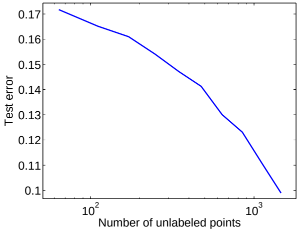
Locally Linear Embedding (LLE)

Roweis and Saul, *Nonlinear dimensionality reduction by locally linear embedding*, Science 2000

$\longrightarrow$ Popular methods for non-linear dimensionality reduction.

- 2D embedding of the 2007 digits of the USPS test set.
- Constructed with a 5 nearest neighbors graph.





- Embedding in 15 dimensions
- Classification by a linear SVM in the embedded space

Outline

1. Introduction
2. Generative models
3. Low density separation
4. Graph based methods
5. Unsupervised learning
6. Conclusions

What do to with unlabeled data ?

1. If the structure contained in the data is irrelevant for the classification problem (i.e. no cluster assumption)
   $\longrightarrow$ Perform standard supervised learning.
2. If you have a good generative model of your data
   $\longrightarrow$ Use it !
3. If the data is clustered and/or high dimensional
   $\longrightarrow$ Use low density separation techniques.
4. If the data has a manifold structure
   $\longrightarrow$ Use a graph based approach.

In all cases, unsupervised learning as a first step is baseline technique that can be very effective.

Benchmark

A lot of variability across methods and datasets

	g241c	g241d	Digit1	USPS	COIL	BCI	Text
1-NN	43.93	42.45	3.89	5.81	17.35	48.67	30.11
SVM	23.11	24.64	5.53	9.75	22.93	34.31	26.45
MVU + 1-NN	43.01	38.20	2.83	6.50	28.71	47.89	32.83
LEM + 1-NN	40.28	37.49	6.12	7.64	23.27	44.83	30.77
QC + CMN	22.05	28.20	3.15	6.36	10.03	46.22	25.71
Discrete Reg.	43.65	41.65	2.77	4.68	9.61	47.67	24.00
TSVM	18.46	22.42	6.15	9.77	25.80	33.25	24.52
SGT	17.41	9.11	2.61	6.80	–	45.03	23.09
Cluster-Kernel	13.49	4.95	3.79	9.68	21.99	35.17	24.38
Entropy-Reg.	20.97	25.36	7.28	12.21	29.48	28.89	24.86
Data-Dep. Reg.	20.31	32.82	2.44	5.10	11.46	47.47	–
LDS	18.04	23.74	3.46	4.96	13.72	43.97	23.15
Laplacian RLS	24.36	26.46	2.92	4.68	11.92	31.36	23.57
CHM (normed)	24.82	25.67	3.79	7.65	–	36.03	–

O. Chapelle, A. Zien, B. Schölkopf, *Semi-Supervised Learning*, MIT Press, 2006

Conclusion

- No "black box" solution: a careful analysis of the problem is needed to understand how the unlabeled can help.
- One of the main challenge is to design large scale algorithms.