Semi-supervised Learning
Introduction

• Supervised learning: \( \{(x^r, \hat{y}^r)\}_{r=1}^R \)
  
  • E.g. \( x^r \): image, \( \hat{y}^r \): class labels

• Semi-supervised learning: \( \{(x^r, \hat{y}^r)\}_{r=1}^R, \{x^u\}_{u=R}^{R+U} \)
  
  • A set of unlabeled data, usually \( U \gg R \)
  
  • Transductive learning: unlabeled data is the testing data
  
  • Inductive learning: unlabeled data is not the testing data

• Why semi-supervised learning?
  
  • Collecting data is easy, but collecting “labelled” data is expensive
  
  • We do semi-supervised learning in our lives
Why semi-supervised learning helps?

Labelled data

Unlabelled data

(Image of cats and dogs without labeling)
Why semi-supervised learning helps?

The distribution of the unlabeled data tell us *something*. Usually with some assumptions.

Who knows?
Outline

Semi-supervised Learning for Generative Model

Low-density Separation Assumption

Smoothness Assumption

Better Representation
Semi-supervised Learning for Generative Model
Supervised Generative Model

• Given labelled training examples \( x^r \in C_1, C_2 \)
  • looking for most likely prior probability \( P(C_i) \) and class-dependent probability \( P(x|C_i) \)
  • \( P(x|C_i) \) is a Gaussian parameterized by \( \mu^i \) and \( \Sigma \)

\[
P(C_1|x) = \frac{P(x|C_1)P(C_1)}{P(x|C_1)P(C_1) + P(x|C_2)P(C_2)}
\]
Semi-supervised Generative Model

• Given labelled training examples $x^r \in C_1, C_2$
  • looking for most likely prior probability $P(C_i)$ and class-dependent probability $P(x|C_i)$
  • $P(x|C_i)$ is a Gaussian parameterized by $\mu^i$ and $\Sigma$

The unlabeled data $x^u$ help re-estimate $P(C_1), P(C_2), \mu^1, \mu^2, \Sigma$
Semi-supervised Generative Model

- Initialization: \( \theta = \{ P(C_1), P(C_2), \mu^1, \mu^2, \Sigma \} \)

- Step 1: compute the posterior probability of unlabeled data

\[
P_\theta(C_1|x^u) \quad \text{Depending on model } \theta
\]

- Step 2: update model

\[
P(C_1) = \frac{N_1 + \sum_{x^u} P(C_1|x^u)}{N}
\]

\[
\mu^1 = \frac{1}{N_1} \sum_{x^r \in C_1} x^r + \frac{1}{\sum_{x^u} P(C_1|x^u)} \sum_{x^u} P(C_1|x^u)x^u \quad \text{......}
\]

The algorithm converges eventually, but the initialization influences the results.

Back to step 1
Why?

• Maximum likelihood with labelled data
  \[
  \log L(\theta) = \sum_{x^r} \log P_\theta(x^r, \hat{y}^r)
  \]
  \[
  = P_\theta(x^r | \hat{y}^r)P(\hat{y}^r)
  \]

  **Closed-form solution**

• Maximum likelihood with labelled + unlabeled data
  \[
  \log L(\theta) = \sum_{x^r} \log P_\theta(x^r, \hat{y}^r) + \sum_{x^u} \log P_\theta(x^u)
  \]

  \[
  P_\theta(x^u) = P_\theta(x^u | C_1)P(C_1) + P_\theta(x^u | C_2)P(C_2)
  \]

  \[(x^u \text{ can come from either } C_1 \text{ and } C_2)\]

\[
\theta = \{P(C_1), P(C_2), \mu^1, \mu^2, \Sigma\}
\]
Semi-supervised Learning
Low-density Separation

非黑即白
“Black-or-white”
Self-training

• Given: labelled data set = \{ (x^r, \hat{y}^r) \}_{r=1}^R, unlabeled data set = \{ x^u \}_{u=l}^{R+U}

• Repeat:
  • Train model $f^*$ from labelled data set
  • Apply $f^*$ to the unlabeled data set
    • Obtain \{ (x^u, y^u) \}_{u=l}^{R+U}
    • Remove a set of data from unlabeled data set, and add them into the labeled data set

Independent to the model
Regression?
Pseudo-label

How to choose the data set remains open
You can also provide a weight to each data.
Self-training

- Similar to semi-supervised learning for generative model
- Hard label v.s. Soft label

Considering using neural network

\( \theta^* \) (network parameter) from labelled data

\[
\begin{bmatrix}
0.7 \\
0.3
\end{bmatrix}
\]

\( x^u \) → \( \theta^* \) →

- Hard
- Soft

70% Class 1
30% Class 2

Class 1 → New target for \( x^u \) is \( \begin{bmatrix} 1 \\ 0 \end{bmatrix} \)

It looks like class 1, then it is class 1.

New target for \( x^u \) is \( \begin{bmatrix} 0.7 \\ 0.3 \end{bmatrix} \)

Doesn’t work ...
Entropy-based Regularization

Entropy of $y^u$:
Evaluate how concentrate the distribution $y^u$ is

$E(y^u) = 0$

$E(y^u) = - \sum_{m=1}^{5} y^u_m ln(y^u_m)$

As small as possible

$E(y^u) = 0$

$E(y^u) = - \sum_{m=1}^{5} y^u_m ln(y^u_m)$

$L = \sum_{x^r} C(y^r, \hat{y}^r)$

labelled data

$\sum_{x^u} E(y^u) + \lambda \sum_{x^u} E(y^u)$

unlabeled data
Outlook: Semi-supervised SVM

Enumerate all possible labels for the unlabeled data

Find a boundary that can provide the largest margin and least error

Thorsten Joachims, "Transductive Inference for Text Classification using Support Vector Machines", ICML, 1999
Semi-supervised Learning
Smoothness Assumption

近朱者赤，近墨者黑
“You are known by the company you keep”
Smoothness Assumption

- Assumption: “similar” $x$ has the same $\hat{y}$
- More precisely:
  - $x$ is not uniform.
  - If $x^1$ and $x^2$ are close in a high density region, $\hat{y}^1$ and $\hat{y}^2$ are the same.

connected by a high density path

$x^1$ and $x^2$ have the same label
$x^2$ and $x^3$ have different labels

Source of image:
http://hips.seas.harvard.edu/files/pinwheel.png
Smoothness Assumption

“indirectly” similar with stepping stones

(The example is from the tutorial slides of Xiaojin Zhu.)

Smoothness Assumption

- Classify astronomy vs. travel articles

(The example is from the tutorial slides of Xiaojin Zhu.)
Smoothness Assumption

- Classify astronomy vs. travel articles

<table>
<thead>
<tr>
<th></th>
<th>$d_1$</th>
<th>$d_5$</th>
<th>$d_6$</th>
<th>$d_7$</th>
<th>$d_3$</th>
<th>$d_4$</th>
<th>$d_8$</th>
<th>$d_9$</th>
<th>$d_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>asteroid</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bright</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>comet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>year</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>zodiac</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>airport</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bike</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>camp</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>yellowstone</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>zion</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(The example is from the tutorial slides of Xiaojin Zhu.)
Cluster and then Label

Using all the data to learn a classifier as usual
Graph-based Approach

• How to know $x^1$ and $x^2$ are close in a high density region (connected by a high density path)

Represented the data points as a **graph**

Graph representation is nature sometimes.

E.g. Hyperlink of webpages, citation of papers

Sometimes you have to construct the graph yourself.
Graph-based Approach - Graph Construction

- Define the similarity $s(x^i, x^j)$ between $x^i$ and $x^j$
- Add edge:
  - K Nearest Neighbor
  - $e$-Neighborhood
- Edge weight is proportional to $s(x^i, x^j)$

Gaussian Radial Basis Function:

$$s(x^i, x^j) = \exp \left( -\gamma \|x^i - x^j\|^2 \right)$$
Graph-based Approach

The labelled data influence their neighbors.
Propagate through the graph
Graph-based Approach

• Define the smoothness of the labels on the graph

\[ S = \frac{1}{2} \sum_{i,j} w_{i,j} (y^i - y^j)^2 \]

Smaller means smoother

For all data (no matter labelled or not)
Graph-based Approach

• Define the smoothness of the labels on the graph

\[ S = \frac{1}{2} \sum_{i,j} w_{i,j} (y^i - y^j)^2 = y^T L y \]

\( y \): (R+U)-dim vector

\[ y = [\ldots y^i \ldots y^j \ldots]^T \]

\( L \): (R+U) x (R+U) matrix

Graph Laplacian

\[ L = D - W \]

\[ W = \begin{bmatrix}
0 & 2 & 3 & 0 \\
2 & 0 & 1 & 0 \\
3 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 
\end{bmatrix} \]

\[ D = \begin{bmatrix}
5 & 0 & 0 & 0 \\
0 & 3 & 0 & 0 \\
0 & 0 & 5 & 0 \\
0 & 0 & 0 & 1 
\end{bmatrix} \]
Graph-based Approach

- Define the smoothness of the labels on the graph

\[ S = \frac{1}{2} \sum_{i,j} w_{i,j} (y^i - y^j)^2 = y^T Ly \]

\[ L = \sum_{x^r} C(y^r, \hat{y}^r) + \lambda S \]

As a regularization term

Depending on network parameters

Semi-supervised Learning
Better Representation

去蕪存菁，化繁為簡
Looking for Better Representation

• Find the latent factors behind the observation
• The latent factors (usually simpler) are better representations

Better representation (Latent factor)
Reference

http://olivier.chapelle.cc/ssl-book/
Acknowledgement

• 感謝 劉議隆 同學指出投影片上的錯字
• 感謝 丁勃雄 同學指出投影片上的錯字