Semi-supervised Learning

Makoto Yamada, Hisashi Kashima
myamada@i.kyoto-u.ac.jp

Kyoto University

July/2/2018

Review: Supervised Learning

Problem formulation of supervised learning.

- Input vector: $\boldsymbol{x} = [x_1, x_2, \ldots, x_d]^\top \in \mathbb{R}^d$
- Output: $y \in \mathbb{R}$
- $(\boldsymbol{x}_i, y_i) \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}, y)$
- Labeled data: $\{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_n, y_n)\}$
- Model: $f(\boldsymbol{x}; \boldsymbol{w}) = \boldsymbol{w}^\top \boldsymbol{x}$. (Linear model)

Risk: $R(\boldsymbol{w}) = \iint \text{loss}(y, f(\boldsymbol{x}; \boldsymbol{w})) p(\boldsymbol{x}, y) \mathrm{d}\boldsymbol{x} \mathrm{d}y$

Empirical Risk: $R_{emp}(\boldsymbol{w}) = \frac{1}{n} \sum_{i=1}^{n} \text{loss}(y_i, f(\boldsymbol{x}_i; \boldsymbol{w}))$

Empirical Risk Minimization (ERM): $\widehat{\boldsymbol{w}} = \operatorname{argmin}_{\boldsymbol{w}} \; R_{emp}(\boldsymbol{w})$

Semi-Supervised Learning

Problem formulation of semi-supervised learning.

- $(\boldsymbol{x}_i, y_i) \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}, y)$
- $\boldsymbol{x}_i \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x})$
- Labeled data: $\{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_n, y_n)\}$
- Unlabeled data: $\{\boldsymbol{x}_{n+1}, \boldsymbol{x}_{n+2}, \ldots, \boldsymbol{x}_{n+m}\}$
- Usually $n \ll m$.

Semi-supervised learning:

- We have both labeled and unlabeled samples.
- Semi-supervised learning uses both labeled and unlabeled samples.
- The unlabeled samples follow the same distribution of the marginal distribution of $p(\boldsymbol{x}, y)$

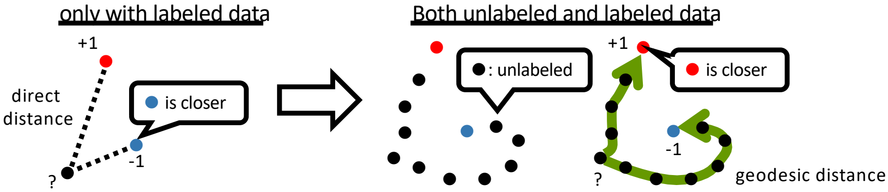
Role of unlabeled data

Data generation process

- Input $\boldsymbol{x}$ is generated by a distribution with probability density $p(\boldsymbol{x})$
- Output y for $\boldsymbol{x}$ is generated by conditional distribution with probability density $p(y|\boldsymbol{x})$.

Unlabeled data can be used for capturing $p(\boldsymbol{x})$

- input data distribution, input space metric, or better representation.



Semi-supervised learning problem: Learning with labeled and unlabeled data

We have both labeled and unlabeled instances (samples):

- Labeled data: $\{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_n, y_n)\}$
- Unlabeled data: $\{\boldsymbol{x}_{n+1}, \boldsymbol{x}_{n+2}, \ldots, \boldsymbol{x}_{n+m}\}$

Estimate a deterministic mapping from $\boldsymbol{x}$ to y.

- Regression
- Classification : $p(y|\boldsymbol{x})$

Typical approaches of semi-supervised learning

- Weighted maximum likelihood estimation
- Graph-based learning
- self-training
- Clustering
- Generative models

Weighted maximum likelihood

The original goal of ML estimation is to maximize:

$$\mathbb{E}_{\boldsymbol{x},y}[\log p(y|\boldsymbol{x})] = \iint \log P(y|\boldsymbol{x};\boldsymbol{w})p(\boldsymbol{x})p(y|\boldsymbol{x})\mathrm{d}\boldsymbol{x}\mathrm{d}y,$$
$$\approx \frac{1}{n}\sum_{i=1}^{n}\log(P(y_i|\boldsymbol{x}_i;\boldsymbol{w}))$$

where $P(y|\boldsymbol{x};\boldsymbol{w})$ is a model. Each training instance is equally weighted.

Note, ML is equivalent to maximize the negative log-likelihood function:

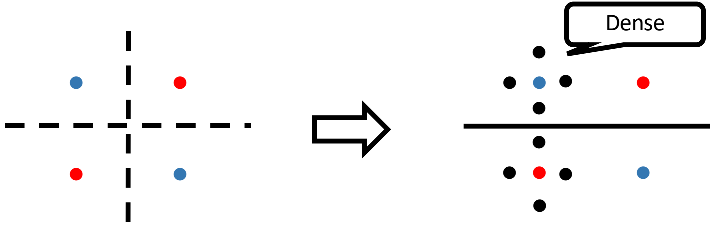
$$L(\boldsymbol{w}) = \log\left(\prod_{i=1}^{n}P(y_i|\boldsymbol{x}_i;\boldsymbol{w})\right)$$
$$\propto \frac{1}{n}\sum_{i=1}^{n}\log(P(y_i|\boldsymbol{x}_i;\boldsymbol{w}))$$

Weighted maximum likelihood

Weighted maximum likelihood:

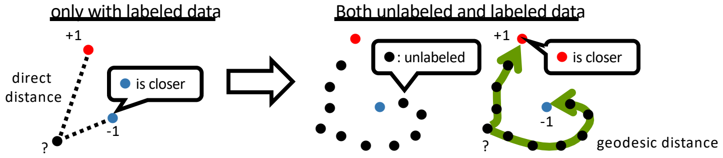
$$\max_{\boldsymbol{w}} \sum_{i=1}^{n} p(\boldsymbol{x}_i) \log(P(y_i | \boldsymbol{x}_i; \boldsymbol{w}))$$

- Each training data instance is weighted by $p(\boldsymbol{x}_i)$.
- $p(\boldsymbol{x})$ is estimated by using unlabeled data.
- Denser areas are largely weighted
- Training a classifier focusing on the dense areas



Graph-based method

- Basic idea: construct a graph capturing the intrinsic shape of input space, and make prediction on the graph.
- Assumption: Data lie on a manifold in the feature space
- The graph represent adjacency relationships among data
- K-nearest neighbor graph (e.g., $A_{ij} = 0, 1$)
- Edge-weighted graph with e.g., $A_{ij} = \exp(-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2^2)$



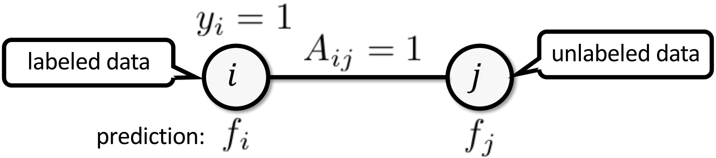
Label propagation

- Basic idea: Adjacent instances tend to have the same label
- Transductive setting (we have test instances)

$$\min_{\boldsymbol{f}\in\mathbb{R}^n} \sum_{i=1}^{n}(f_i - y_i)^2 + \lambda \sum_{i,j} A_{ij}(f_i - f_j)^2,$$

where $\lambda > 0$ is the regularization parameter.

- 1st term: (squared) loss function to fit to labeled data.
- 2nd term: regularization function to make adjacent nodes to have similar predictions.



Illustrative example of label propagation

Predict if people are infected by some disease

- Test results are known for some people
- infections spread over social networks