Unleash the Power of Label Space: Label Enhancement

Xin Geng

PAttern Learning and Mining (PALM) Lab
http://palm.seu.edu.cn

School of Computer Science and Engineering
Southeast University

Outline

- Introduction
- A Theoretical View
- Label Enhancement Methods
  - Fuzzy Label
  - Probabilistic Label
  - Label Distribution
- Applications
- Conclusion

0/1 Labels

- Most existing data sets: a bipartition of the label set into relevant and irrelevant labels
  - 1： relevant label
  - 0： irrelevant label

0/1 Labels

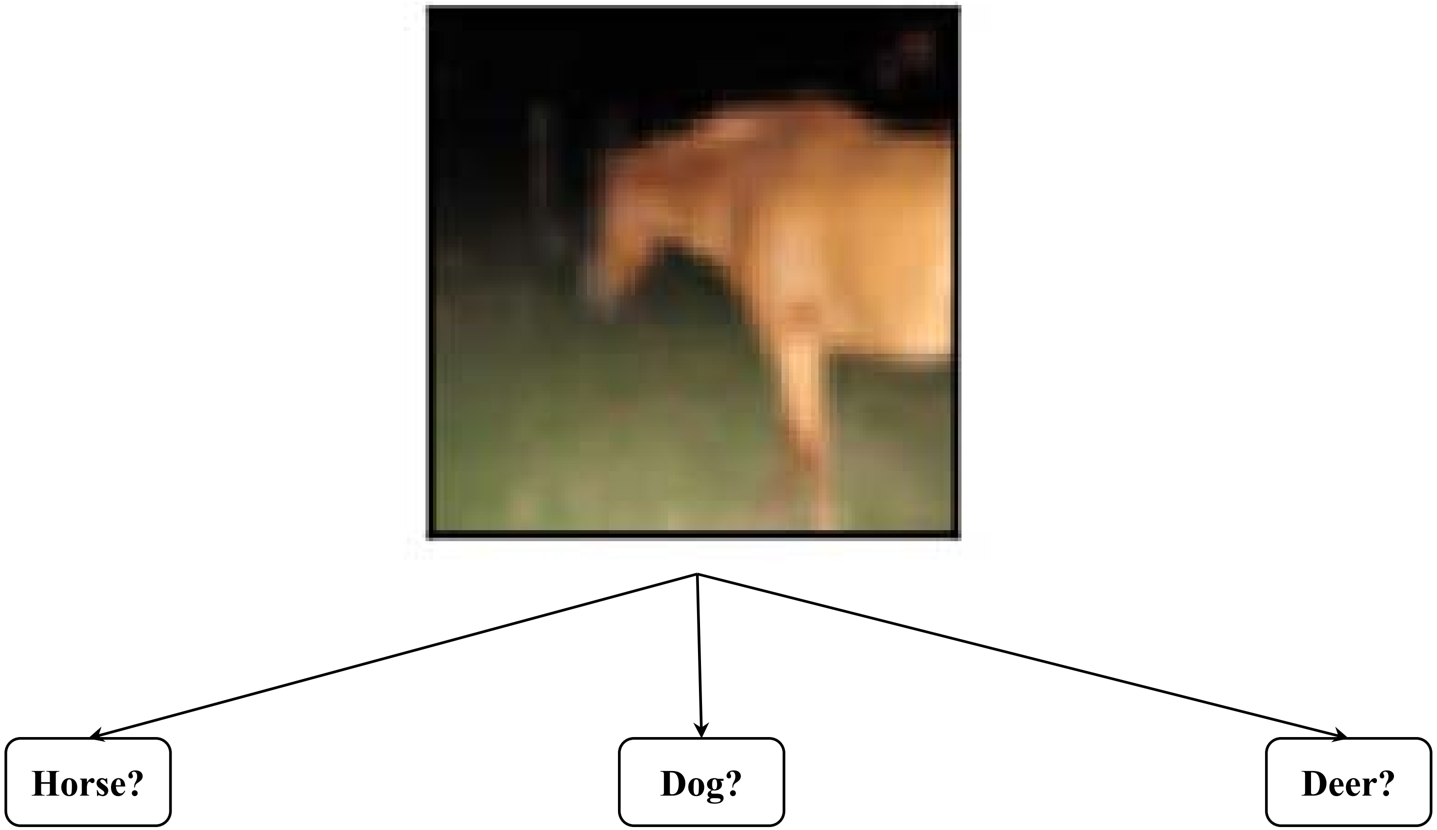
咖啡	图书	钢笔	大象
1	1	0	0

Fuzziness

maturity
rare
medium rare
medium
medium well
well done

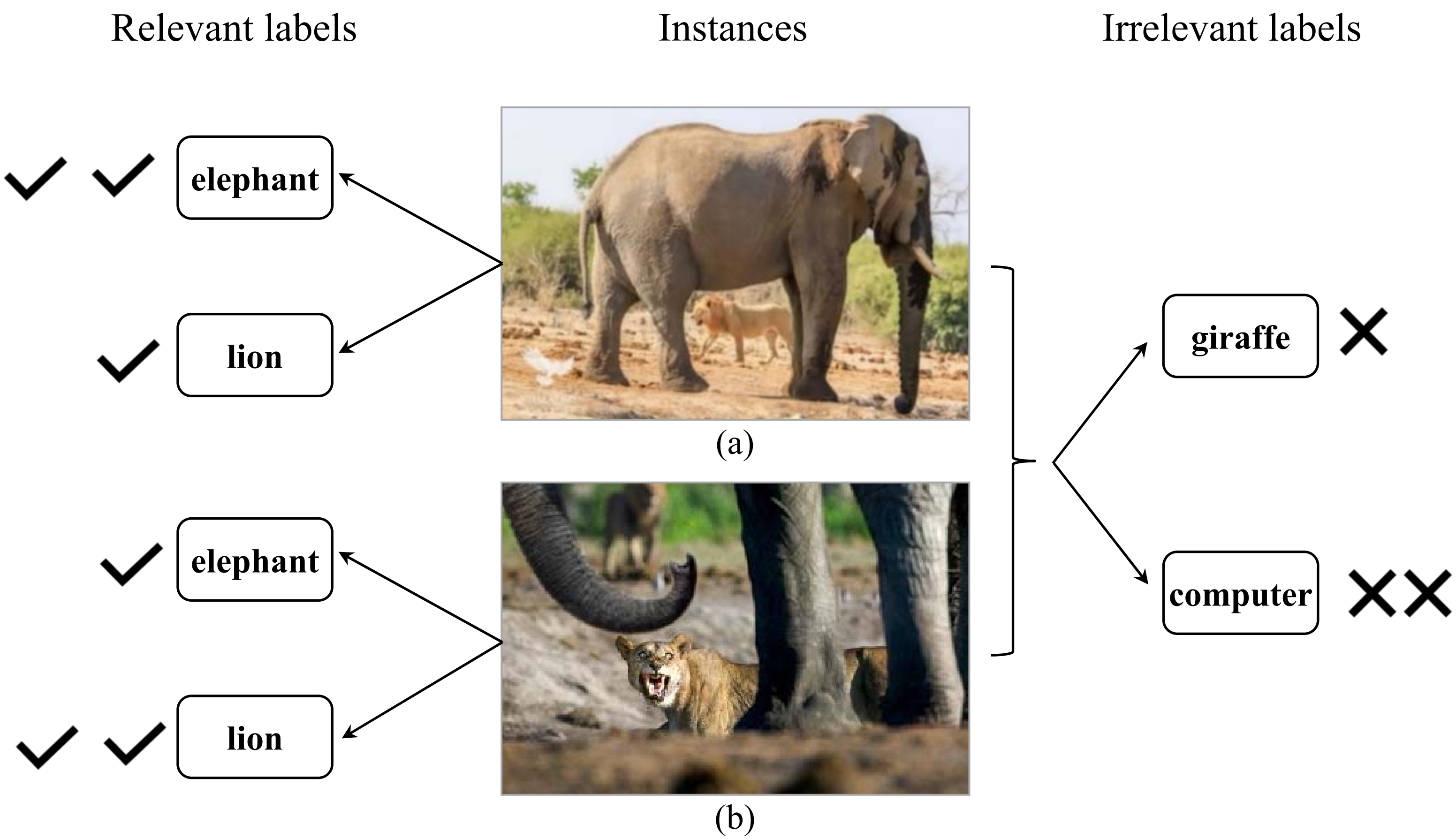
The definition of the class labels is fuzzy

Probability



The label relevance is probabilistic

Ambiguity



Class labels of the instance are ambiguous

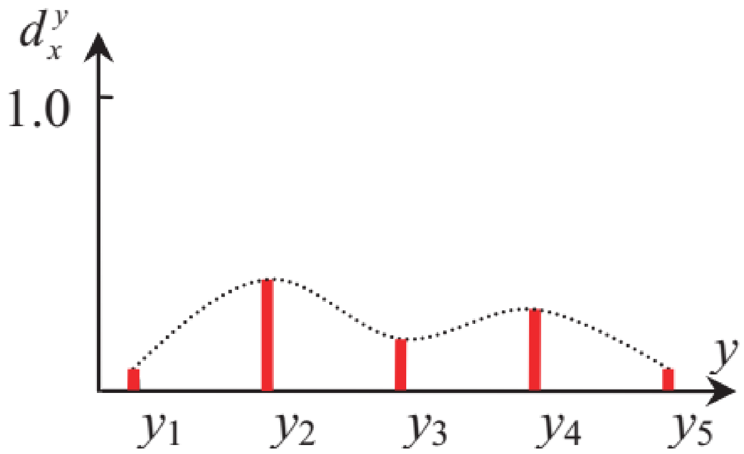
0/1 Label → Fine Label

A real number $d_{\boldsymbol{x}}^{y}$ is assigned to the label y for the instance $\boldsymbol{x}$

WLOG ⇒ $d_{\boldsymbol{x}}^{y} \in [0,1]$

Complete label set ⇒ $\sum_y d_{\boldsymbol{x}}^{y} = 1$

} Fine label

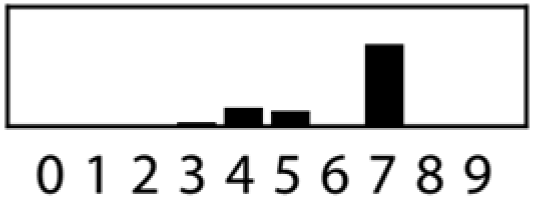


Fuzziness – Membership Degree

maturity	membership degree
rare	0.05
medium rare	0.2
medium	0.25
medium well	0.35
well done	0.15

The definition of the class labels is fuzzy

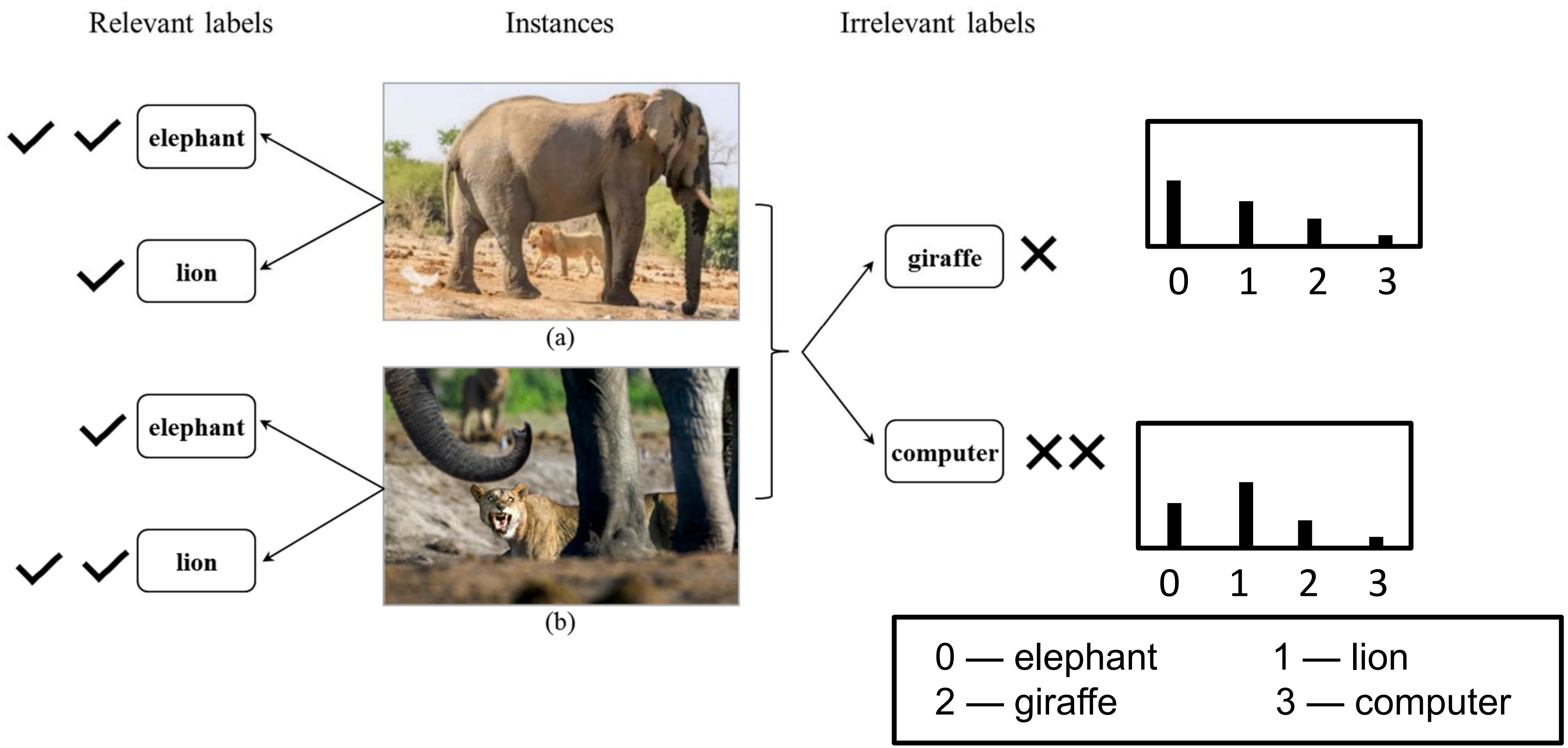
Probability – Probabilistic Labels



0 — airplane	2 — bird	4 — deer	6 — frog	8 — ship
1 — automobile	3 — cat	5 — dog	7 — horse	9 — truck

The label relevance is probabilistic

Ambiguity – Label Distribution



Class labels of the instance are ambiguous

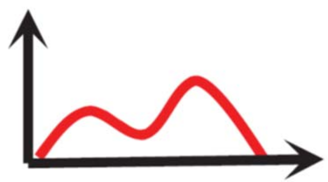
Practical Restrictions

- Directly obtaining fine labels is difficult:
  - **High cost**
  - **Difficult to quantify**
- 0/1 Labels simplify the real world: a bipartition of the label set into relevant and irrelevant labels
  - 1： relevant label
  - 0： irrelevant label

We need a way to **recover** the fine labels from the 0/1 labels in the training set

Label Enhancement (LE)

{0, 1, 0, 1, 0}



Problem Formulation

The 0/1 label vector of $\boldsymbol{x}_i$ is denoted by $\boldsymbol{l}_i = \left(l_{\boldsymbol{x}_i}^{y_1}, l_{\boldsymbol{x}_i}^{y_2}, \ldots, l_{\boldsymbol{x}_i}^{y_c}\right)^{\mathrm{T}}$, where $l_{\boldsymbol{x}_i}^{y_j} \in \{0,1\}$ represents whether y_j describes $\boldsymbol{x}_i$, c is the number of labels. Then, $\boldsymbol{l}_i \in \{0,1\}^c$.

The fine label vector of $\boldsymbol{x}_i$ is denoted by $\boldsymbol{d}_i = \left(d_{\boldsymbol{x}_i}^{y_1}, d_{\boldsymbol{x}_i}^{y_2}, \ldots, d_{\boldsymbol{x}_i}^{y_c}\right)^{\mathrm{T}}$, where $d_{\boldsymbol{x}_i}^{y_j} \in [0,1]$ represents the fine label of y_j to $\boldsymbol{x}_i$. Then, $\boldsymbol{d}_i \in [0,1]^c$.

Label Enhancement can be defined as follows.

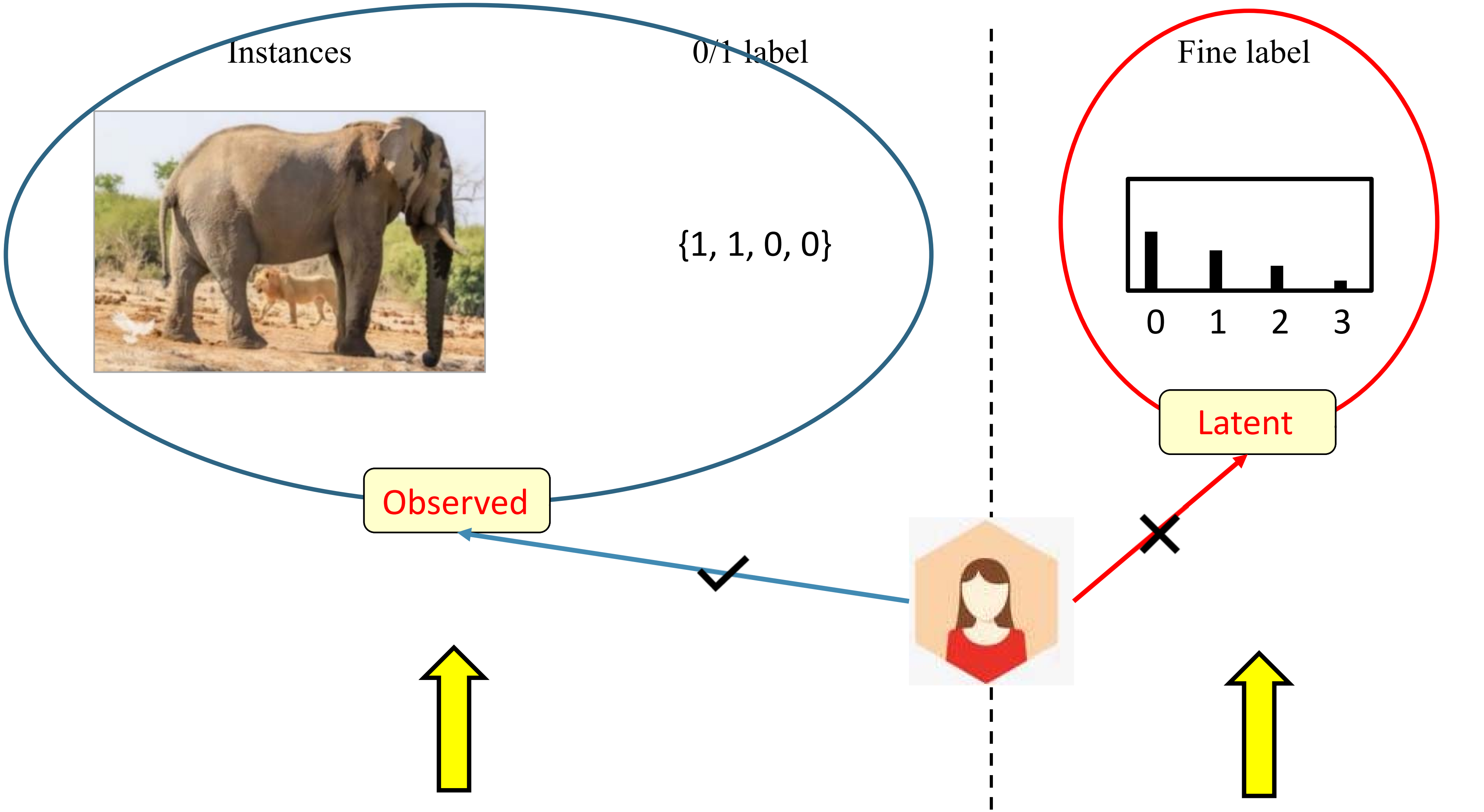
Given a training set $S = \{(\boldsymbol{x}_i, \boldsymbol{l}_i) | 1 \leq i \leq n\}$, label enhancement is to recover the fine label vector $\boldsymbol{d}_i$ of $\boldsymbol{x}_i$ from the 0/1 label vector $\boldsymbol{l}_i$, and thus transform S into a fine label training set $E = \{(\boldsymbol{x}_i, \boldsymbol{d}_i) | 1 \leq i \leq n\}$.

Outline

- **Introduction**
- **A Theoretical View**
- **Label Enhancement Methods**
  - **Fuzzy Label**
  - **Probabilistic Label**
  - **Label Distribution**
- **Applications**
- **Conclusion**

A Theoretical View

[Xu and Geng, ICML'20]

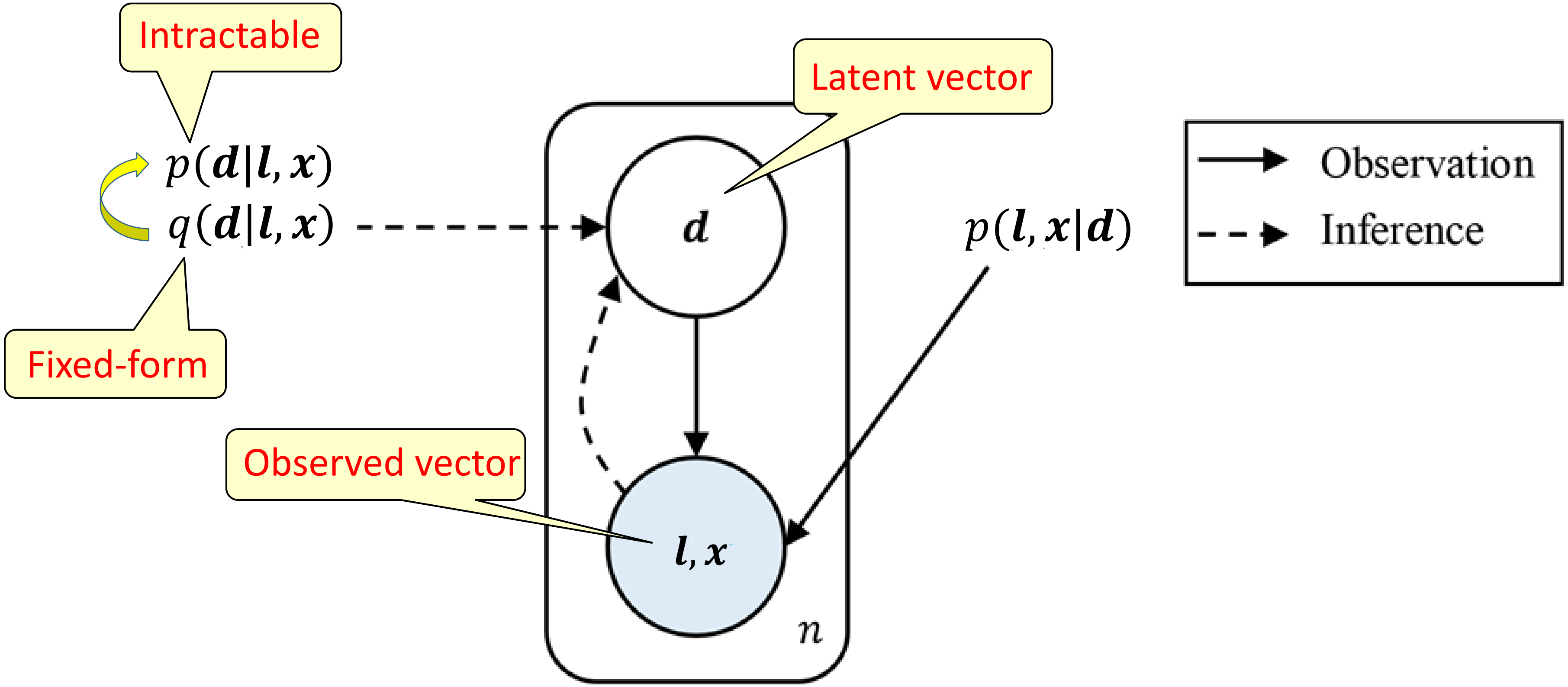


conditional distribution $p(\boldsymbol{l}, \boldsymbol{x}|\boldsymbol{d})$

posterior distribution $p(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})$

A Theoretical View

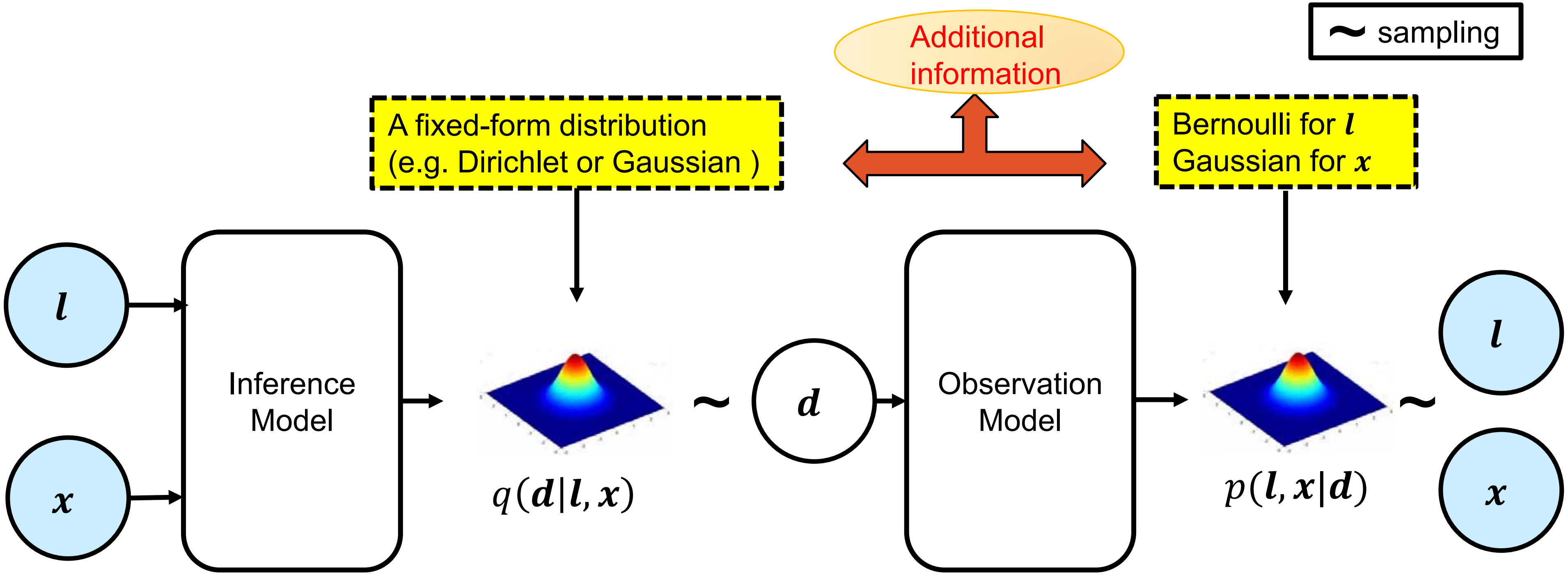
[Xu and Geng, ICML'20]



- $\boldsymbol{l}$, $\boldsymbol{x}$ are generated from some conditional distribution $p(\boldsymbol{l},\boldsymbol{x}|\boldsymbol{d})$.
- $\boldsymbol{d}$ is generated from the posterior distribution $p(\boldsymbol{d}|\boldsymbol{l},\boldsymbol{x})$.
- A fixed-form distribution $q(\boldsymbol{d}|\boldsymbol{l},\boldsymbol{x})$ is utilized to approximate $p(\boldsymbol{d}|\boldsymbol{l},\boldsymbol{x})$.

A Theoretical View

[Xu and Geng, ICML'20]



$\boldsymbol{d}$ should maximize the lower bound of the joint probability density $p(\boldsymbol{l},\boldsymbol{x})$:

$$\log p(\boldsymbol{l},\boldsymbol{x}) \geq \mathbb{E}_{q(\boldsymbol{d}|\boldsymbol{l},\boldsymbol{x})}[\log p(\boldsymbol{l}|\boldsymbol{d}) + \log p(\boldsymbol{x}|\boldsymbol{d})] - \mathrm{KL}[q(\boldsymbol{d}|\boldsymbol{l},\boldsymbol{x})||p(\boldsymbol{d})]$$

A Theoretical View

[Xu and Geng, ICML'20]

We formulate the label enhancement problem into an optimization framework and yields the target function for minimization:

$$T(\boldsymbol{\vartheta}, \boldsymbol{\eta}, \boldsymbol{w}) = \frac{1}{L} \sum_{m=1}^{L} \frac{1}{2} \|\boldsymbol{x} - \boldsymbol{\rho}^{(m)}\|_2^2 - \sum_{i=1}^{c} l_i \log \tau_i^{(m)} + (1 - l_i) \cdot \log \left(1 - \tau_i^{(m)}\right)$$
$$+ \frac{1}{2}\{\mathrm{tr}(\boldsymbol{\Sigma}) + \boldsymbol{\mu}^\top \boldsymbol{\mu} - k - \log |\boldsymbol{\Sigma}|\},$$

where $\boldsymbol{\Sigma} = \mathrm{MLP}_{\boldsymbol{\Sigma}}(\boldsymbol{l}, \boldsymbol{x}; \boldsymbol{w})$, $\boldsymbol{\mu} = \mathrm{MLP}_{\boldsymbol{\mu}}(\boldsymbol{l}, \boldsymbol{x}; \boldsymbol{w})$, $\boldsymbol{\tau}^{(m)} = \mathrm{MLP}_{\boldsymbol{\tau}}(\boldsymbol{d}^{(m)}; \boldsymbol{\vartheta})$,
$\boldsymbol{\rho}^{(m)} = \mathrm{MLP}_{\boldsymbol{\rho}}(\boldsymbol{d}^{(m)}; \boldsymbol{\eta})$, $\boldsymbol{d}^{(m)} = \boldsymbol{\mu} + \boldsymbol{\Sigma}\boldsymbol{\epsilon}^{(m)}$, $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.

Outline

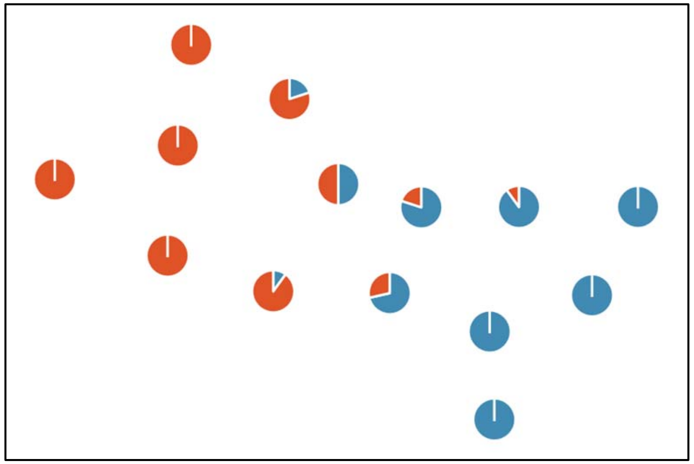
- **Introduction**
- **A Theoretical View**
- **Label Enhancement Methods**
  - **Fuzzy Label**
  - **Probabilistic Label**
  - **Label Distribution**
- **Applications**
- **Conclusion**

Label Enhancement Methods

- **Label Enhancement for Fuzzy Label**
  - LE based on fuzzy clustering
    [Gayar et al., ANNPR'06]
  - LE based on kernel method
    [Jiang et al., NCA'06]
- **Label Enhancement for Probabilistic Label**
  - LE based on probabilistic end-to-end noisy correction
    [Kun Yi et al., CVPR'19]
  - LE based on label smoothing
    [Szegedy et al., CVPR'16]
  - LE based on distillation
    [Hinton et al., arxiv'15]
- **Label Enhancement for Label Distribution**
  - LE based on manifold learning
    [Hou et al., AAAI'16]
  - Graph laplacian label enhancement
    [Xu and Geng, IJCAI'18]
  - LE based on reinforcement learning
    [Gao and Geng, IJCAI'20]

LE based on fuzzy clustering

[Gayar et al., ANNPR'06]



Fuzzy C-means clustering
(The membership of the instance to the cluster)

The memberships of the instances belonging to the same class are added up to form the cluster-class connection matrix.

By fuzzy composition operation, the memberships of instances to clusters are transformed into the memberships of instances to class labels using the connection matrix.

LE based on fuzzy clustering

[Gayar et al., ANNPR'06]

- Step 1: Fuzzy C-Means clustering (FCM)
  1. Given the cluster number p, initialize the $n \times p$ cluster membership matrix $\boldsymbol{M}$ (m_{ik} denotes the membership of $\boldsymbol{x}_i$ to the k-th cluster)
  2. Calculate the cluster prototype
     $$\boldsymbol{\mu}_k = \frac{\sum_{i=1}^{n}(m_{ik})^{\beta}\boldsymbol{x}_i}{\sum_{i=1}^{n}(m_{ik})^{\beta}}$$
  3. Update the cluster membership matrix $\boldsymbol{M}$
     $$m_{ik} = \frac{1}{\sum_{j=1}^{p}\left(\frac{Dist(\boldsymbol{x}_i,\boldsymbol{\mu}_k)}{Dist(\boldsymbol{x}_i,\boldsymbol{\mu}_j)}\right)^{\frac{2}{\beta-1}}}$$
  4. Repeat 2 and 3 until convergence

Each row of $\boldsymbol{M}$, $\boldsymbol{m}_i$, represents the membership of the instance $\boldsymbol{x}_i$ to each cluster

LE based on fuzzy clustering [Gayar et al., ANNPR'06]

- Step 2: Calculate the cluster-class connection matrix
  1. Initialize $c \times p$ zero matrix $\boldsymbol{A}$
  2. Update each row $\boldsymbol{A}_j$ with

$$\boldsymbol{A}_j = \boldsymbol{A}_j + \boldsymbol{m}_i, \qquad if\ l_{\boldsymbol{x}_i}^{y_j} = 1$$

  3. Normalized each column of $\boldsymbol{A}$
  4. Normalized each row of $\boldsymbol{A}$

a_{jk} denotes the connection between class j and cluster k

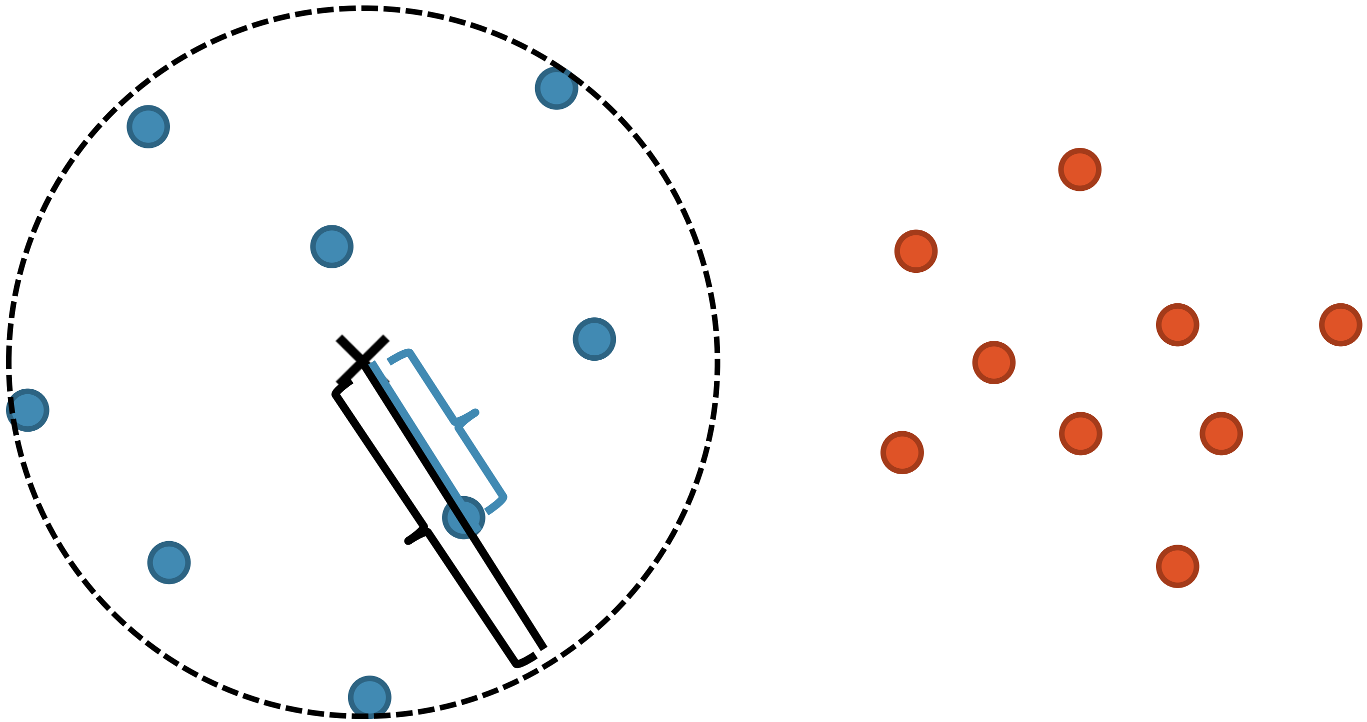
- Step 3: Calculate the fine labels of $\boldsymbol{x}_i$
  1. $\boldsymbol{D}_i = \boldsymbol{A} \circ \boldsymbol{m}_i$ (fuzzy composition)

$$D_i^j = \max_k (a_{jk} \times m_{ik})$$

  2. Normalize $\boldsymbol{D}_i$

LE based on kernel method

[Jiang et al., NCA'06]



Introduce nonlinearity via kernel method

LE based on kernel method

[Jiang et al., NCA'06]

- Step 1: For each label y_j, suppose C^{y_j} contains all the instances labeled by y_j, the size of C^{y_j} is n_j, then, the center of C^{y_j} is

$$\boldsymbol{\Psi}^{y_j} = \frac{1}{n_j}\sum_{\boldsymbol{x}_i \in C^{y_j}} \phi(\boldsymbol{x}_i)$$

$(\boldsymbol{\Psi}^{y_j})^2$ can be calculated via inner product of $\phi(\boldsymbol{x}_i)$

where $\phi(\boldsymbol{x}_i)$ is a nonlinear function determined by the kernel function

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \phi(\boldsymbol{x}_i) \cdot \phi(\boldsymbol{x}_j)$$

- Step 2: Calculate the class radius

$$r_j = \max_{\boldsymbol{x}_i \in C^{y_j}} \|\boldsymbol{\Psi}^{y_j} - \phi(\boldsymbol{x}_i)\|,$$

r_j^2 can be calculated via inner product of $\phi(\boldsymbol{x}_i)$

- Step 3: Calculate the distance between instance $\boldsymbol{x}_i$ and class center

$$d_{ij}^2 = \|\phi(\boldsymbol{x}_i) - \boldsymbol{\Psi}^{y_j}\|^2$$

d_{ij}^2 can be calculated via inner product of $\phi(\boldsymbol{x}_i)$

LE based on kernel method

[Jiang et al., NCA'06]

- Step 4: calculate the membership of instance $\boldsymbol{x}_i$ to label y_j

$$m_{\boldsymbol{x}_i}^{y_j} = \begin{cases} 1 - \sqrt{\dfrac{\left\| d_{ij}^2 \right\|}{\left(r_j^2 + \delta\right)}} & if\ l_{\boldsymbol{x}_i}^{y_j} = 1 \\ 0 & if\ l_{\boldsymbol{x}_i}^{y_j} = 0 \end{cases}$$

Cannot change the membership of irrelevant labels

- Step 5: Normalize $\boldsymbol{m}_{\boldsymbol{x}_i} = \left[m_{\boldsymbol{x}_i}^{y_1}, m_{\boldsymbol{x}_i}^{y_2}, \ldots, m_{\boldsymbol{x}_i}^{y_c}\right]$

Outline

- **Introduction**
- **A Theoretical View**
- **Label Enhancement Methods**
  - **Fuzzy Label**
  - **Probabilistic Label**
  - **Label Distribution**
- **Applications**
- **Conclusion**

LE based on probabilistic end-to-end noisy correction

[Kun Yi et al., CVPR'19]

Step 1: Probabilistic label is initialized by the noisy label $\hat{\boldsymbol{y}}$

$$\tilde{\boldsymbol{y}} = K\hat{\boldsymbol{y}}$$

K is a large constant

Step 2: Normalize the probabilistic label to a probability distribution

$$\boldsymbol{y}^d = softmax(\tilde{\boldsymbol{y}})$$

Step 3: Update both the network and the probabilistic label $\boldsymbol{y}^d$

LE based on probabilistic end-to-end noisy correction

[Kun Yi et al., CVPR'19]

Loss 1: Compatibility loss $\mathcal{L}_o$, using the useful information in noisy label $\widehat{\boldsymbol{Y}}$

$$\mathcal{L}_o\left(\widehat{\boldsymbol{Y}}, \boldsymbol{Y}^d\right) = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{c}\widehat{\boldsymbol{y}}_{ij}\log \boldsymbol{y}_{ij}^d$$

Loss 2: Classification loss $\mathcal{L}_c$, update both network and probabilistic label

$$\mathcal{L}_c\left(f(\boldsymbol{x};\boldsymbol{\theta}), \boldsymbol{Y}^d\right) = \frac{1}{n}\sum_{i=1}^{n}\mathrm{KL}(f(\boldsymbol{x}_i;\boldsymbol{\theta})||\boldsymbol{y}_i^d)$$

Loss 3: Entropy loss $\mathcal{L}_e$, a regularization term to force the network to peak at only one category rather than being flat

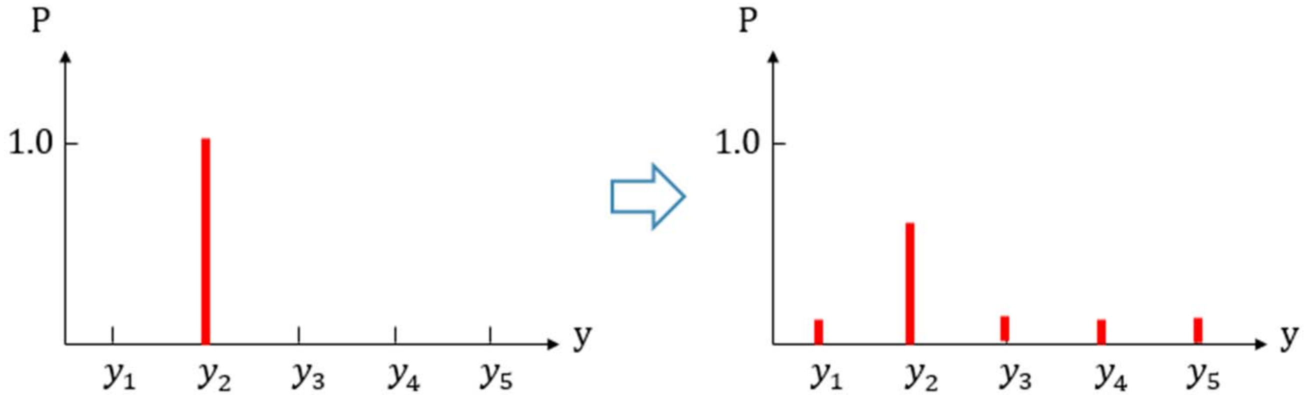
$$\mathcal{L}_e(f(\boldsymbol{x};\boldsymbol{\theta})) = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{c}f_j(\boldsymbol{x};\boldsymbol{\theta})\log f_j(\boldsymbol{x};\boldsymbol{\theta})$$

The over all framework

$$\mathcal{L} = \frac{1}{n}\mathcal{L}_c\left(f(\boldsymbol{x};\boldsymbol{\theta}), \boldsymbol{Y}^d\right) + \alpha\mathcal{L}_o\left(\widehat{\boldsymbol{Y}}, \boldsymbol{Y}^d\right) + \frac{\beta}{c}\mathcal{L}_e(f(\boldsymbol{x};\boldsymbol{\theta}))$$

LE based on label smoothing

[Szegedy et al., CVPR'16]



- Step 1： After adding label smoothing, the true probability distribution changes as follows

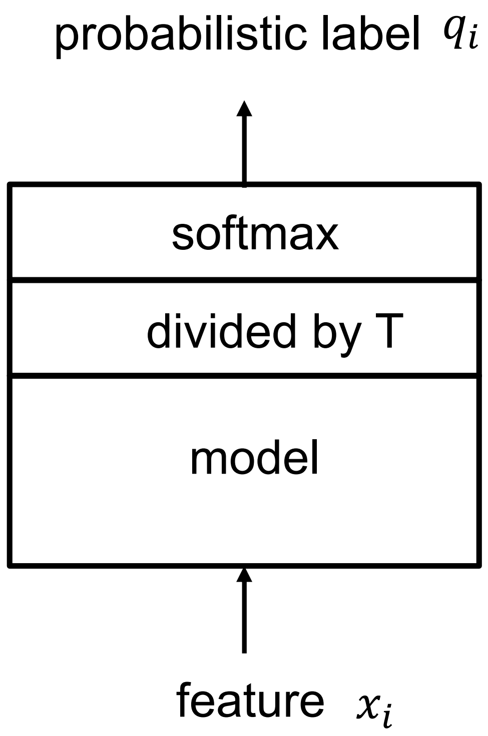
$$P_i = \begin{cases} 1, if\,(i = y) \\ 0, if\,(i \neq y) \end{cases} \Rightarrow P_i = \begin{cases} 1 - \varepsilon, (i = y) \\ \dfrac{\varepsilon}{K-1}, if\,(i \neq y) \end{cases}$$

K = # labels

- Step 2： After adding label smoothing, the loss function changes as follows

$$Loss = -\sum_{i=1}^{K} p_i log q_i \Rightarrow Loss_i = \begin{cases} (1-\varepsilon) * Loss, if\,(i = y) \\ \varepsilon * Loss, if\,(i \neq y) \end{cases}$$

LE based on knowledge distillation [Hinton et al., arxiv'15]



Neural networks typically produce class probabilities by using a "softmax" output layer that converts the logit, z_i , computed for each class into a probability, q_i, by comparing z_i with the other logits.

$$q_i = \frac{\exp\left(\frac{z_i}{T}\right)}{\sum_j \exp\left(\frac{z_j}{T}\right)}$$

T is a temperature that is normally set to 1. Using a higher value for T produces a softer probability distribution over classes.

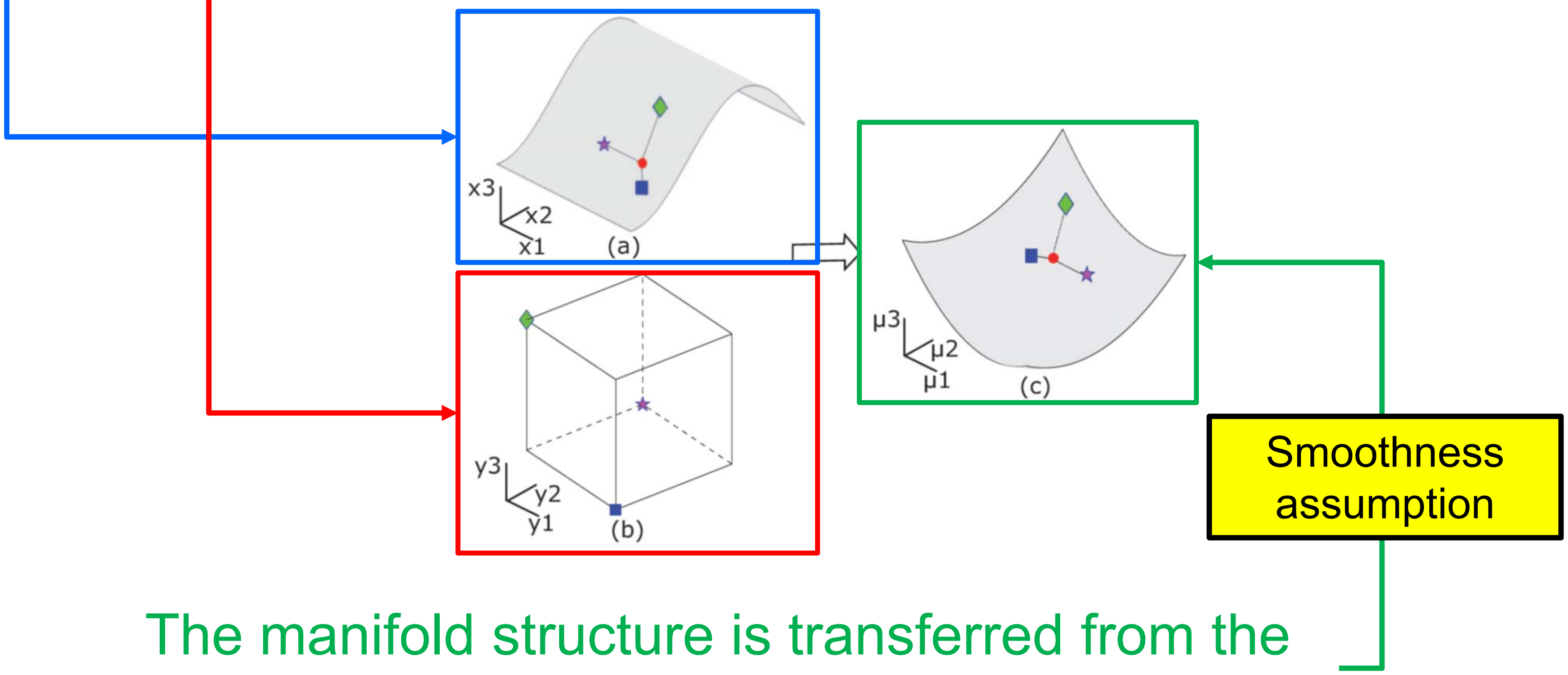
Outline

- **Introduction**
- **A Theoretical View**
- **Label Enhancement Methods**
  - **Fuzzy Label**
  - **Probabilistic Label**
  - **Label Distribution**
- **Applications**
- **Conclusion**

LE based on manifold learning

[Hou, Geng and Zhang, AAAI'16]

- Feature space： continuous Euclidean space
- Label space： discrete logical space



The manifold structure is transferred from the feature space to the label space.

LE based on manifold learning

[Hou, Geng and Zhang, AAAI'16]

- Manifold learning in feature space [Roweis & Saul, Science, 2000]

$$\arg\min_{\boldsymbol{W}} \sum_{i=1}^{n} \|\boldsymbol{x}_i - \sum_{j \neq i} W_i^j \boldsymbol{x}_j\|^2$$

$$\text{s.t.} \quad \mathbf{1}^{\mathrm{T}} \boldsymbol{W}_i = 1$$

Local topological structure

- Manifold learning in label space

$$\arg\min_{\boldsymbol{\mu}} \sum_{i=1}^{n} \|\boldsymbol{\mu}_i - \sum_{j \neq i} W_i^j \boldsymbol{\mu}_j\|^2$$

$$\text{s.t.} \quad \forall 1 \leq i \leq n, 1 \leq l \leq q \qquad y_i^l \mu_i^l \geq \lambda, \lambda > 0$$

Graph Laplacian Label Enhancement

[Xu and Geng, IJCAI'18]

- Model

Nonlinear transformation

$$\boldsymbol{D}_i = \boldsymbol{W}^\top \varphi(\boldsymbol{x}_i) + \boldsymbol{b} = \widehat{\boldsymbol{W}} \boldsymbol{\phi}_i$$

Goal Determining the best parameter $\widehat{\boldsymbol{W}}^*$

- Target function

Logical label loss

$$\min_{\widehat{\boldsymbol{W}}} L(\widehat{\boldsymbol{W}}) + \lambda \Omega(\widehat{\boldsymbol{W}})$$

Feature space constraint

Graph Laplacian Label Enhancement

[Xu and Geng, IJCAI'18]

- The first part of the target function

$$L(\widehat{\boldsymbol{W}}) = \sum_{i=1}^{n} \left\| \widehat{\boldsymbol{W}} \boldsymbol{\phi}_i - \boldsymbol{L}_i \right\|^2$$

Least squares (LS)

- The second part of the target function

Smoothness assumption

$$\Omega(\widehat{\boldsymbol{W}}) = \sum_{i,j} a_{ij} \left\| \boldsymbol{D}_i - \boldsymbol{D}_j \right\|^2 = \boldsymbol{tr}(\boldsymbol{D}\boldsymbol{G}\boldsymbol{D}^{\top})$$

$$a_{ij} = \begin{cases} \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\sigma^2}\right) & \text{if } \boldsymbol{x}_j \in N(i) \\ 0 & \text{otherwise} \end{cases}$$

Correlation between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$

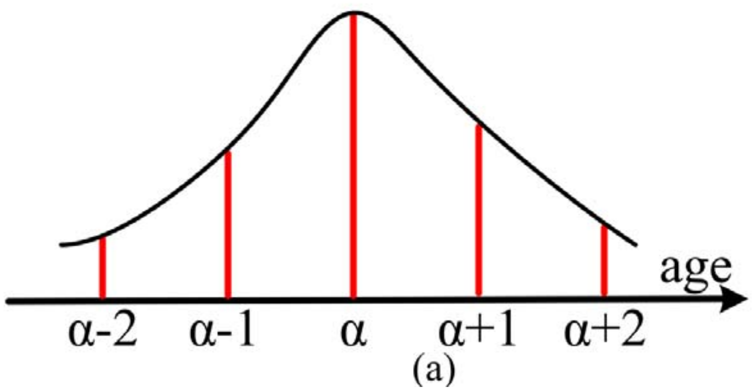
Graph Laplacian

$$\boldsymbol{G} = \widehat{\boldsymbol{A}} - \boldsymbol{A},\ \hat{a}_{ij} = \sum_{j=1}^{n} a_{ij}$$

LE based on reinforcement learning

[Gao and Geng, IJCAI’20]

- Leveraging the prior knowledge



Prior knowledge in Age Estimation
[Geng, et al. TPAMI, 2013]

Prior knowledge in Emotion Relation
[Mikels, et al. BRM, 2005]

The properties implied in the prior knowledge

1. $d_x^{\alpha} > d_x^{\beta}$, $\alpha \neq \beta$
2. $d_x^{\alpha \pm i} > d_x^{\alpha \pm j}$, $j > i$, $i, j > 0$

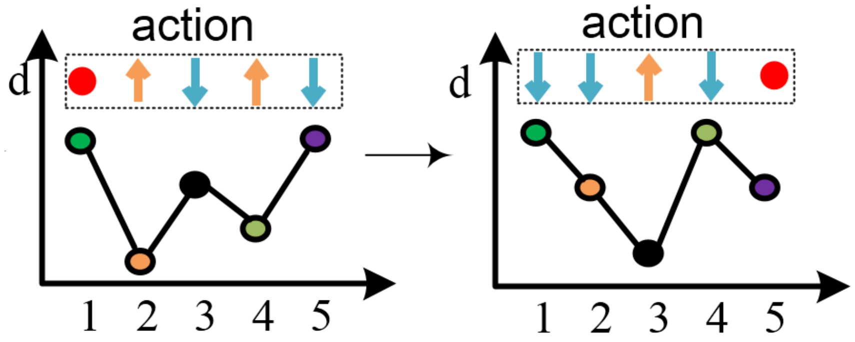
ground-truth hard label

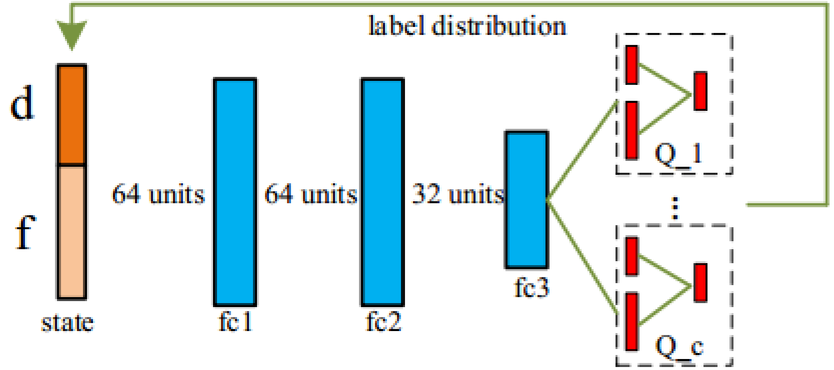
1. $d_x^{y_g} > d_x^{y_i}$, $i \neq g$
2. $d_i^{y_i} > d_i^{y_j}$, $distance(j, g) > distance(i, g)$

LE based on reinforcement learning

[Gao and Geng, IJCAI'20]

- Reinforcement learning for LE





a constant value

$$d_x^{y_i} = d_x^{y_i} + \delta, \; d_x^{y_i} = d_x^{y_i} - \delta, \; d_x^{y_i} = d_x^{y_i}$$

$$MQ(s,a) = \sum_i^c Q_i(s,a;\theta,\phi,\beta),$$

$$d_x^{y_i} = \frac{e^{d_x^{y_i}}}{\sum_c e^{d_x^{y_i}}}$$

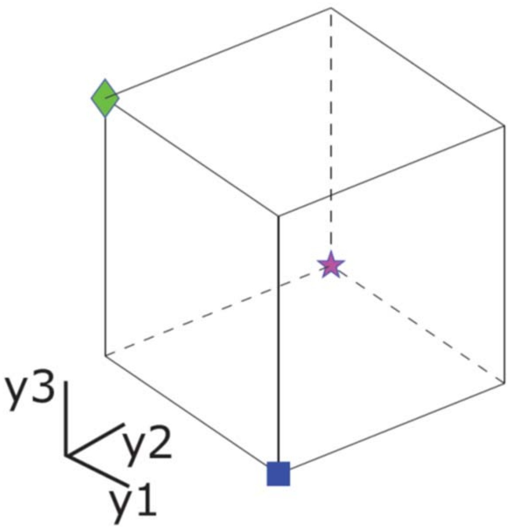
the sum of description degrees is 1.

$$Q(s,a;\theta,\phi,\beta) = V(s;\theta,\beta) + \left(A(s,a;\theta,\phi) - \frac{1}{|A|}\sum_{a'} A(s,a';\theta,\phi)\right),$$

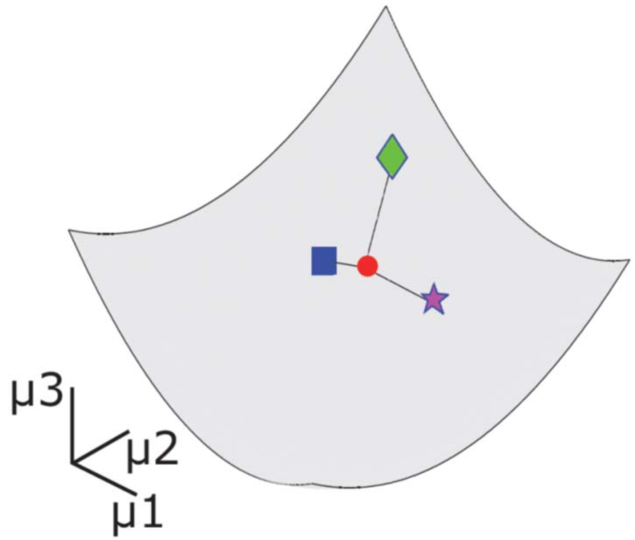
Outline

- Introduction
- A Theoretical View
- Label Enhancement Methods
  - Fuzzy Label
  - Probabilistic Label
  - Label Distribution
- Applications
- Conclusion

The Power of Label Space



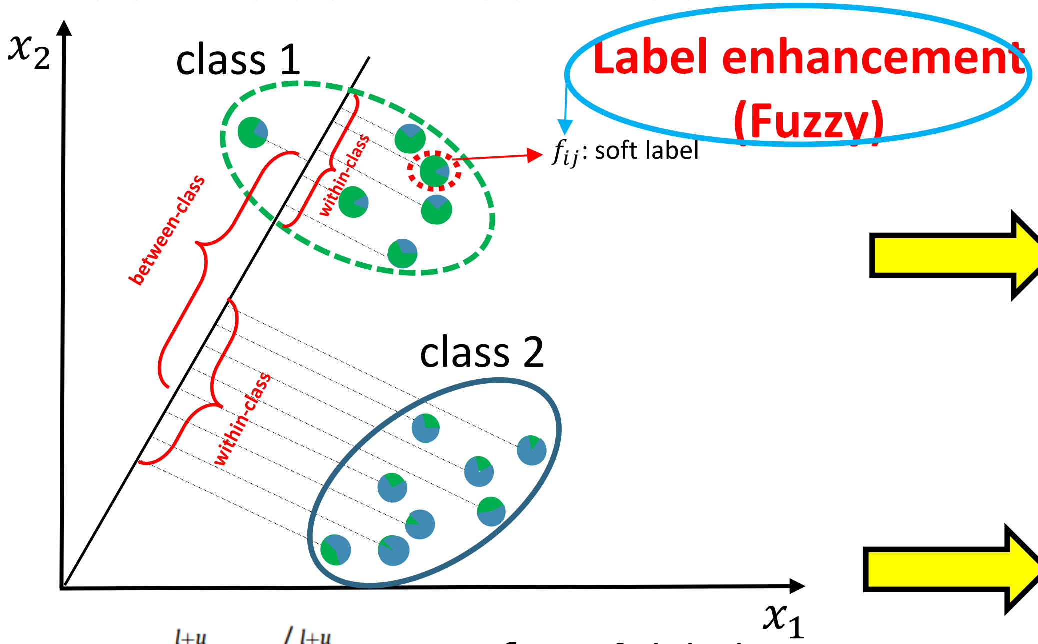
0/1 Label Space



Fine Label Space

Linear Discriminant Analysis [Zhao et al., CVIU'14]

- Soft Label Linear Discriminant Analysis (SL-LDA)

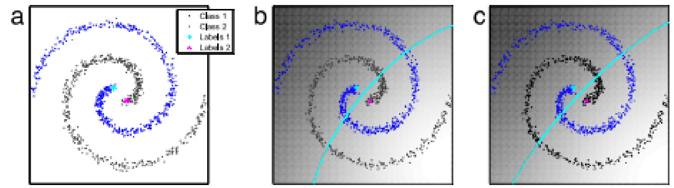


$$\widetilde{\mu}_i = \sum_{j=1}^{l+u} f_{ij}x_j \Big/ \sum_{j=1}^{l+u} f_{ij}$$

f_{ij}: soft label

$$\widetilde{\mu} = \sum_{i=1}^{c}\sum_{j=1}^{l+u} f_{ij}x_j \Big/ \sum_{i=1}^{c}\sum_{j=1}^{l+u} f_{ij}.$$

$$J(V) = \max_{V} Tr((V^T S_w V)^{-1} V S_b V)$$



Dimensionality Reduction
[Zhao et al.. Neural Networks.2014]

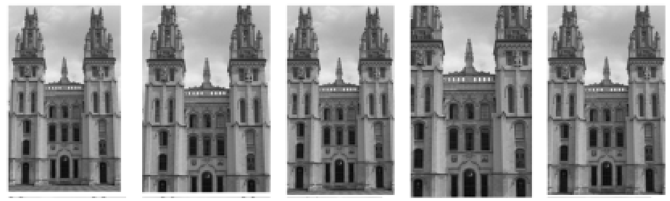


Image Retrieval
[Sovann et al. Wireless Communications Networking Conference. 2015]

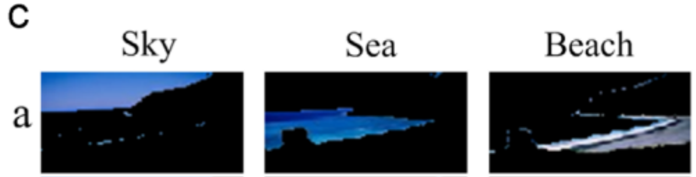


Image Classification
[Zhao et al. Neurocomputing, 2015]

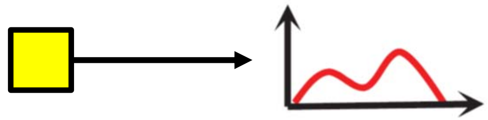
Label Distribution Learning

[Xu and Geng, IJCAI'18]

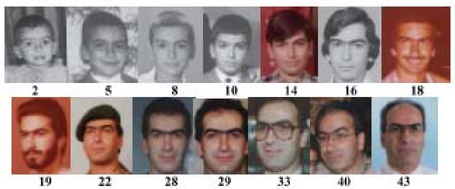
{0, 1, 0, 1, 0}

0/1 label data

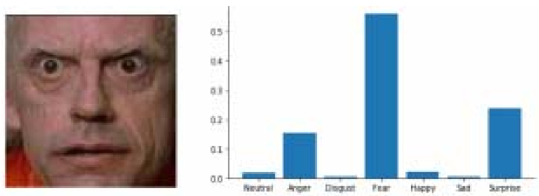
Label enhancement
(Label distribution)



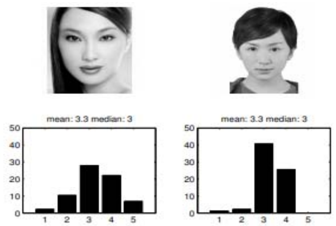
LDL



Age Estimation
[Gao B B et al. TIP, 2016]



Facial Expression Recognition
[Chen et al, CVPR 2020]

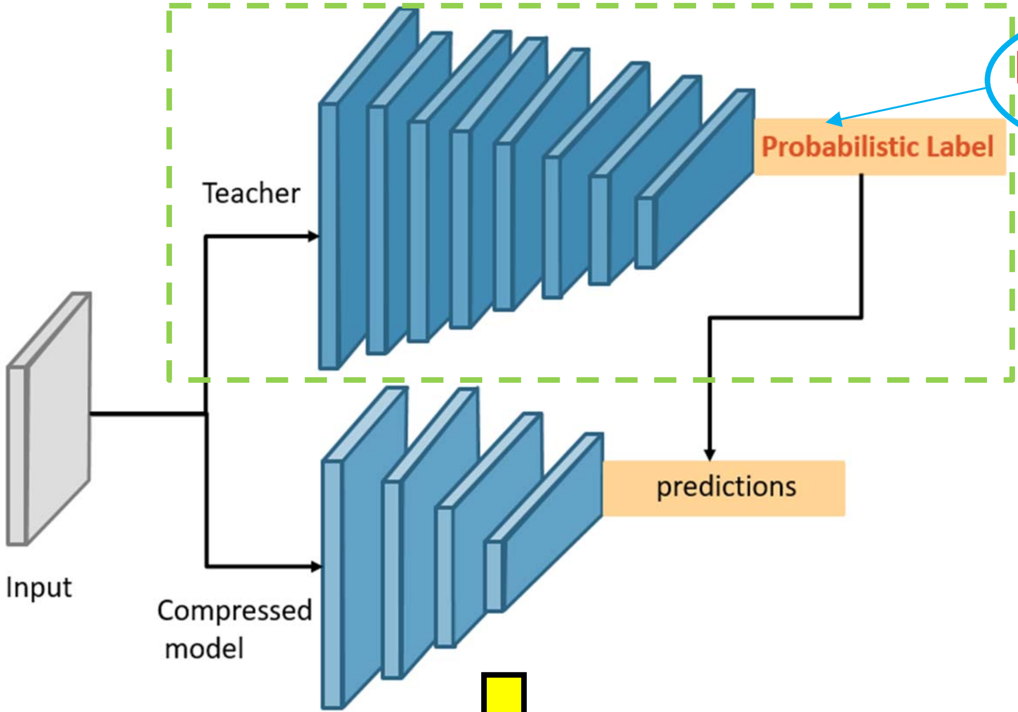


Sense Beauty
[Yi et al, IJCAI, 2017]

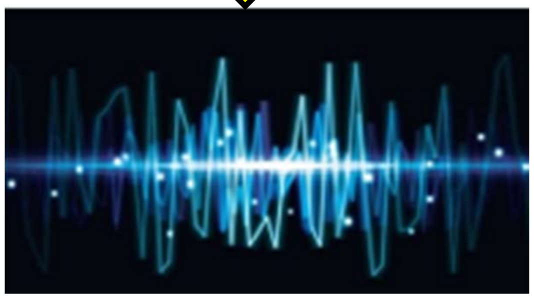
Model Compression

[Hinton et al., arxiv'15]

Label enhancement (Distillation)



Semantic image segmentation
[Chen et al. ECCV, 2018]

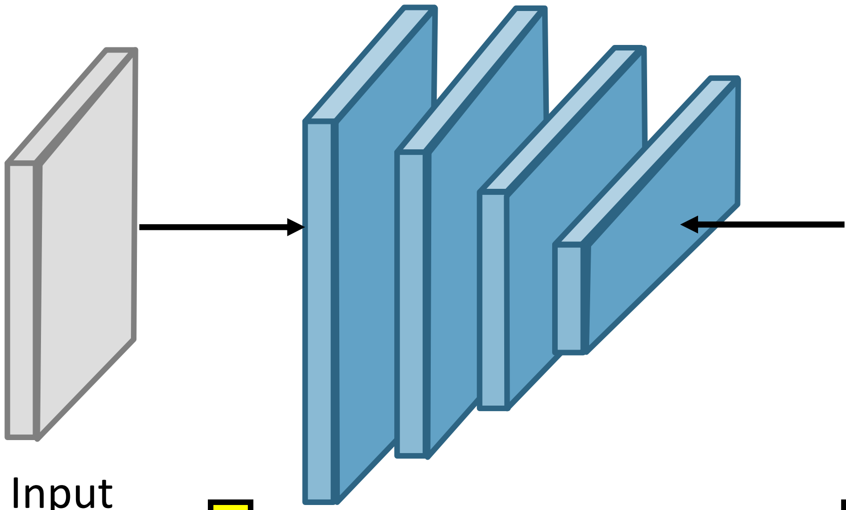


Speech Recognition
[Chebotar et al. Interspeech, 2016]

Mobile vision applications
[Howard et al. arXiv, 2017]

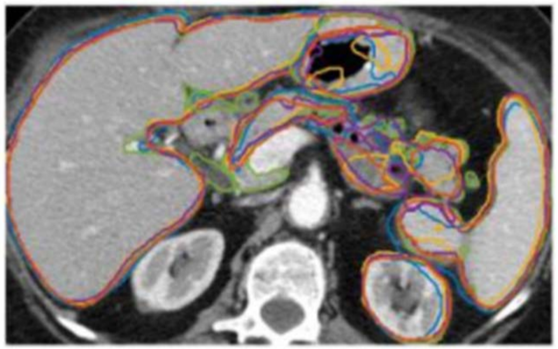
Neural network regularization

[Hinton et al., ICLR'17]

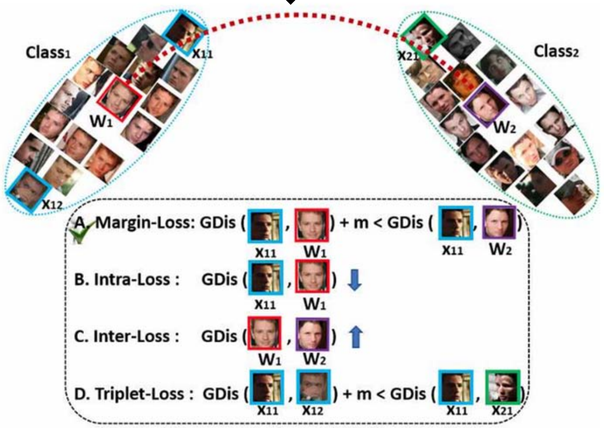


$$P_i = \begin{cases} 1-\varepsilon, (i = y) \\ \dfrac{\varepsilon}{K-1}, if\,(i \neq y) \end{cases}$$

Label enhancement (Label smoothing)



CT segmentation
[Gibson et al. Medical Imaging, 2018]



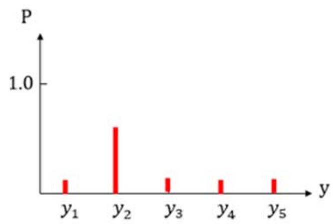
Face recognition
[Deng et al. CVPR, 2019]

Video summarization
[Zhou et al. arXiv, 2018]

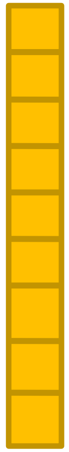
Label embedding

[Sun et al., arxiv'17]

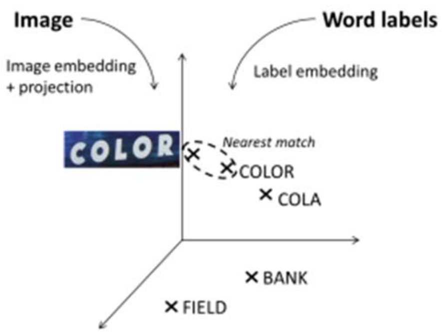
Label enhancement



probabilistic labels



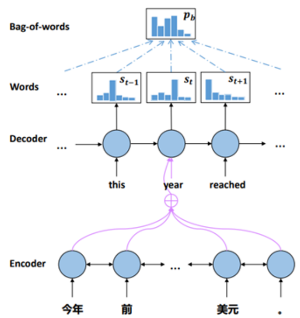
Label embedding



Text recognition
Jose et al. BMVC, 2018.

• Generating descriptions for videos has many applications including human robot interaction.
• Many methods for image captioning rely on pre-trained object classifier CNN and Long Short Term Memory recurrent networks.
• How to learn robust visual classifiers from the weak annotations of the sentence descriptions.

Sequence generation
Yang et al. arXiv, 2018.



Machine translation
Ma et al. arXiv, 2018.

Outline

- **Introduction**
- **A Theoretical View**
- **Label Enhancement Methods**
  - **Fuzzy Label**
  - **Probabilistic Label**
  - **Label Distribution**
- **Applications**
- **Conclusion**

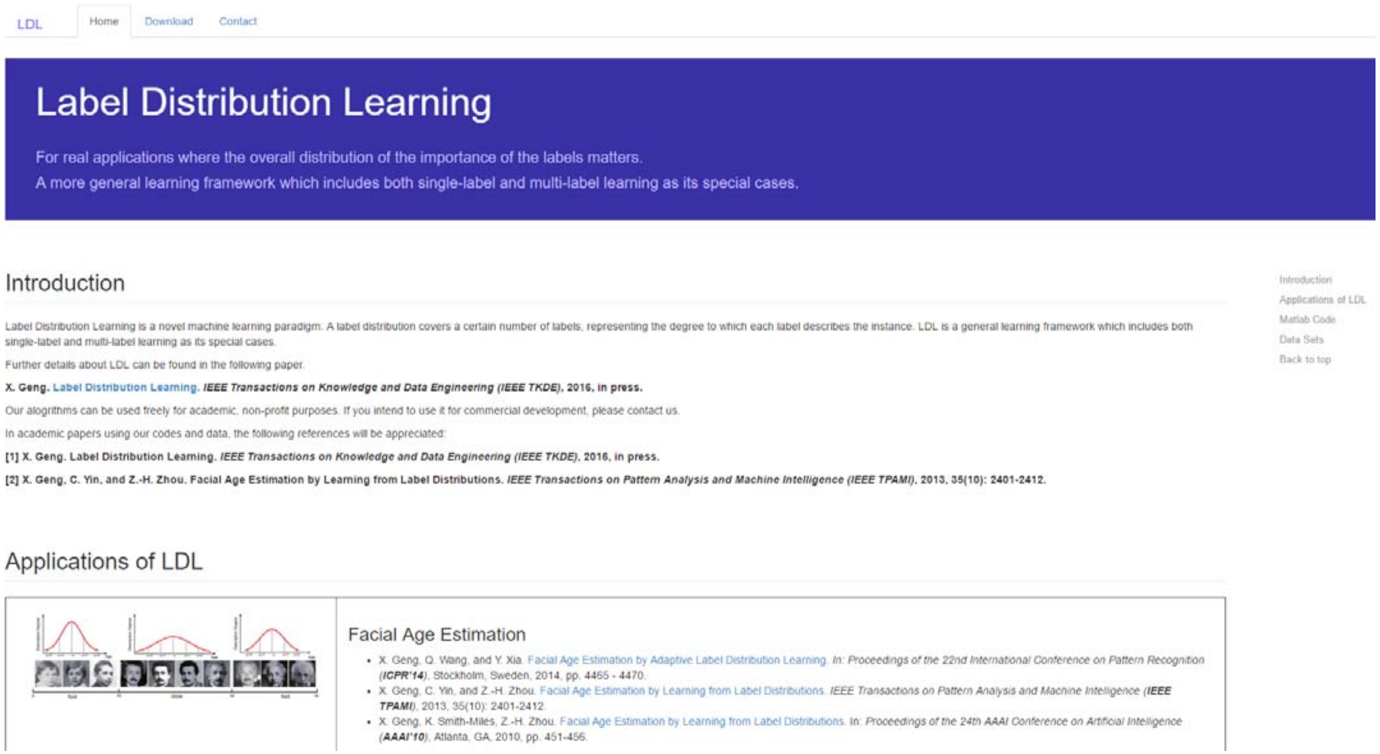
Conclusion

- **Label enhancement**
  - recovers fine labels (e.g., label distribution, probabilistic labels, fuzzy labels) from 0/1 labels.
  - could be theoretically explained via variational inference.
  - offers more possibilities for operations in the label space (e.g., linear discriminant analysis, label distribution learning, model compression, neural network regularization, label embedding).

Interested in LDL & LE?

All the papers、codes and datasets are available at:

http://palm.seu.edu.cn/xgeng/LDL/index.htm


LDL Home Download Contact

Label Distribution Learning

For real applications where the overall distribution of the importance of the labels matters.
A more general learning framework which includes both single-label and multi-label learning as its special cases.

Introduction

Label Distribution Learning is a novel machine learning paradigm. A label distribution covers a certain number of labels, representing the degree to which each label describes the instance. LDL is a general learning framework which includes both single-label and multi-label learning as its special cases.

Further details about LDL can be found in the following paper.

X. Geng. Label Distribution Learning. *IEEE Transactions on Knowledge and Data Engineering (IEEE TKDE)*, 2016, in press.

Our alogrithms can be used freely for academic, non-profit purposes. If you intend to use it for commercial development, please contact us.

In academic papers using our codes and data, the following references will be appreciated:

[1] X. Geng. Label Distribution Learning. *IEEE Transactions on Knowledge and Data Engineering (IEEE TKDE)*, 2016, in press.

[2] X. Geng, C. Yin, and Z.-H. Zhou. Facial Age Estimation by Learning from Label Distributions. *IEEE Transactions on Pattern Analysis and Machine Intelligence (IEEE TPAMI)*, 2013, 35(10): 2401-2412.

Introduction
Applications of LDL
Matlab Code
Data Sets
Back to top

Applications of LDL

Facial Age Estimation

- X. Geng, Q. Wang, and Y. Xia. Facial Age Estimation by Adaptive Label Distribution Learning. *In: Proceedings of the 22nd International Conference on Pattern Recognition (**ICPR'14**)*, Stockholm, Sweden, 2014, pp. 4465 - 4470.
- X. Geng, C. Yin, and Z.-H. Zhou. Facial Age Estimation by Learning from Label Distributions. *IEEE Transactions on Pattern Analysis and Machine Intelligence (**IEEE TPAMI**)*, 2013, 35(10): 2401-2412.
- X. Geng, K. Smith-Miles, Z.-H. Zhou. Facial Age Estimation by Learning from Label Distributions. In: *Proceedings of the 24th AAAI Conference on Artificial Intelligence (**AAAI'10**)*, Atlanta, GA, 2010, pp. 451-456.

THANK YOU！

http:// palm.seu.edu.cn