Transfer Metric Learning by Learning Task Relationships

Yu Zhang and Dit-Yan Yeung

Department of Computer Science and Engineering
Hong Kong University of Science and Technology

KDD 2010
1. **Introduction**

2. **Multi-Task Metric Learning**

3. **Transfer Metric Learning by Learning Task Relationships**

4. **Experiments**

5. **Conclusion**
Metric Learning and Its Limitations

- Distance metric plays a very crucial role in many data mining algorithms.
  - $k$-means clustering, $k$-nearest neighbor classifier, ...

- Its limitation:
  - With only limited labeled data, the metric learned is often unsatisfactory.

- Solutions:
  - Semi-Supervised Metric Learning
    - Utilize information in unlabeled data
  - Transfer Metric Learning
    - Utilize information in other related tasks
Transfer learning is to improve the performance of the target task with the help of some source tasks.
Transfer Metric Learning

There is only one work on transfer metric learning

- [ref]: Robust distance metric learning with auxiliary knowledge, IJCAI’09.

Some Limitations:

- It only models positive task correlation
- The optimization problem is non-convex

Our Contributions:

- Propose a convex formulation for transfer metric learning
- Model the pairwise task relationships under the regularization framework
  - Positive task correlation
  - Negative task correlation
  - Task unrelatedness
Outline

1. Introduction
2. Multi-Task Metric Learning
3. Transfer Metric Learning by Learning Task Relationships
4. Experiments
5. Conclusion
Notations

- $m$ learning tasks $\{T_i\}_{i=1}^m$
- The training set $\mathcal{D}_i$ in $T_i$ consists of $n_i$ data points $(x^i_j, y^i_j)$, $j = 1, \ldots, n_i$
- $x^i_j \in \mathbb{R}^d$ and its corresponding class label $y^i_j \in \{1, \ldots, C_i\}$. 
The Objective Function

- The optimization problem for multi-task metric learning is formulated as follows:

  \[
  \min_{\{\Sigma_i\}, \Omega} \sum_{i=1}^{m} \frac{2}{n_i(n_i - 1)} \sum_{j < k} g\left( y_{j,k} \left[ 1 - \| x_i^j - x_i^k \| \right] \right) + \frac{\lambda_1}{2} \sum_{i=1}^{m} \| \Sigma_i \|_F^2 + \frac{\lambda_2}{2} \text{tr} (\tilde{\Sigma} \Omega^{-1} \tilde{\Sigma}^T) \\
  \text{s.t.} \Sigma_i \succeq 0 \ \forall i \\
  \tilde{\Sigma} = (\text{vec}(\Sigma_1), \ldots, \text{vec}(\Sigma_m)) \\
  \Omega \succeq 0, \ \text{tr}(\Omega) = 1.
  \]

- From the probabilistic viewpoint, this is related to MAP solution of a probabilistic model where the prior on the metrics of all tasks is matrix-variate normal distribution.

- It has been proved that the optimization problem is a convex optimization problem.

- We propose an alternating method to solve the problem efficiently.
Outline

1. Introduction

2. Multi-Task Metric Learning

3. Transfer Metric Learning by Learning Task Relationships

4. Experiments

5. Conclusion
The Assumption

- Suppose we are given $m - 1$ source tasks $\{T_i\}_{i=1}^{m-1}$ and one target task $T_m$.
- Each source task has enough labeled data and can learn an accurate model with no need to seek help from the other source tasks.
- We assume that the metric matrix $\Sigma_i$ for the $i$th source task has been learned independently.
The Objective Function

- Based on multi-task metric learning, we formulate the optimization problem as follows:

\[
\min_{\Sigma_m, \Omega} \frac{2}{n_m(n_m - 1)} \sum_{j < k} g\left( y_{j,k}^m \left[ 1 - \|x_j^m - x_k^m\|^2_{\Sigma_m} \right] \right) + \frac{\lambda_1}{2} \|\Sigma_m\|_F^2 + \frac{\lambda_2}{2} \text{tr}(\tilde{\Sigma} \Omega^{-1} \tilde{\Sigma}^T)
\]

s.t. \( \Sigma_m \succeq 0 \)
\[
\tilde{\Sigma} = (\text{vec}(\Sigma_1), \ldots, \text{vec}(\Sigma_{m-1}), \text{vec}(\Sigma_m))
\]
\[
\Omega \succeq 0, \quad \text{tr}(\Omega) = 1.
\]

- Since we assume that the source tasks are independent and of equal importance, we can express \( \Omega \) as

\[
\Omega = \begin{pmatrix}
\frac{1-\omega}{m-1} \mathbf{1}_{m-1}^T & \omega_m \\
\omega_m & \omega
\end{pmatrix}.
\]
Optimization Procedure

- It can be proved that the problem is jointly convex with respect to all variables: $\Sigma_m$, $\omega_m$ and $\omega$.
- However, it is not easy to optimize it with respect to all the variables simultaneously.
- We still use an alternating method to solve it.
The optimization problem with respect to $\Sigma_m$ is formulated as

$$\min_{\Sigma_m} \frac{2}{n_m(n_m-1)} \sum_{j<k} g\left(y_{j,k}^m \left[ 1 - \|x_j^m - x_k^m\|^2_{\Sigma_m} \right] \right)$$

$$+ \frac{\lambda'_1}{2} \|\Sigma_m\|_F^2 - \lambda'_2 \text{tr}(\Sigma_m^T M)$$

s.t. $\Sigma_m \succeq 0$.

Similar to regularized distance metric learning method, we use an online algorithm to solve this problem.
Input: labeled data $(x_j^m, y_j^m)$ ($j = 1, \ldots, n_m$), matrix $M$, $\lambda'_1$, $\lambda'_2$ and predefined learning rate $\eta$

Initialize $\Sigma_m^{(0)} = \frac{\lambda'_2}{\lambda'_1} M$;

for $t = 1, \ldots, T_{max}$ do

Receive a pair of training data points $\{(x_j^m, y_j^m), (x_k^m, y_k^m)\}$;

Compute $y$: $y = 1$ if $y_j^m = y_k^m$, and $y = -1$ otherwise;

if the training pair $(x_j^m, x_k^m, y)$ is classified correctly, i.e., $y(1 - \|x_j^m - x_k^m\|^2 \Sigma_m^{(t-1)}) > 0$ then

$$\Sigma_m^{(t)} = \Sigma_m^{(t-1)};$$

else if $y = -1$

$$\Sigma_m^{(t)} = \Sigma_m^{(t-1)} + \eta(x_j^m - x_k^m)(x_j^m - x_k^m)^T;$$

else

$$\Sigma_m^{(t)} = \pi_{S_+} \left( \Sigma_m^{(t-1)} - \eta(x_j^m - x_k^m)(x_j^m - x_k^m)^T \right)$$

where $\pi_{S_+}(A)$ projects matrix $A$ into the positive semidefinite cone;

end if

end for

Output: metric $\Sigma_m^{(T_{max})}$
Optimization Procedure - Optimizing w.r.t. $\omega_m$ and $\omega$

- The optimization problem with respect to $\omega_m$ and $\omega$ is formulated as

$$\min_{\omega_m, \omega, \Omega} \text{tr}(\tilde{\Sigma} \Omega^{-1} \tilde{\Sigma}^T)$$

subject to

$$\Omega = \begin{pmatrix} \frac{1-\omega}{m-1} I_{m-1} & \omega_m \\ \omega_m^T & \omega \end{pmatrix}$$

$$\omega(1 - \omega) \geq (m - 1) \omega_m^T \omega_m.$$

- Then we can reformulate it as a second-order cone programming problem:

$$\min_{\omega_m, \omega, f, t, \{h_j\}, \{r_j\}} -t$$

subject to

$$\frac{1 - \omega}{m - 1} \geq t \lambda_1, \quad f = U^T (\omega_m - t \Psi_{12})$$

$$r_j = \frac{1 - \omega}{m - 1} - t \lambda_j, \quad \left\| \begin{pmatrix} f_j \\ r_j - h_j \end{pmatrix} \right\|_2 \leq \frac{r_j + h_j}{2} \quad \forall j$$

$$\sum_{j=1}^{m-1} h_j \leq \omega - t \Psi_{22}, \quad \left\| \begin{pmatrix} \sqrt{m-1} \omega_m \\ \omega - 1 \\ 2 \omega \end{pmatrix} \right\|_2 \leq \frac{\omega + 1}{2}.$$
Outline

1. Introduction
2. Multi-Task Metric Learning
3. Transfer Metric Learning by Learning Task Relationships
4. Experiments
5. Conclusion
Experimental Setup

- **Three baseline methods** are compared:
  - Information-Theoretic Metric Learning (ITML)
    - [ref]: Information-theoretic metric learning, ICML’07.
  - Regularized distance metric learning (RDML)
    - [ref]: Regularized distance metric learning: Theory and algorithm, NIPS’09.
  - Existing transfer metric learning method - LDML
    - [ref]: Robust distance metric learning with auxiliary knowledge, IJCAI’09.

- **CVX solver** is used to solve the second-order cone programming problem.

- The learning rate $\eta$ is set to be 0.01.
Wine Quality Classification

- This is to classify wine into different grades from 0 to 10.
- There are two tasks:
  - One for red wine classification
  - The other for white wine classification
- Each task is treated as the target task and the other task as the source task.
- To see the effect of varying the size of the training set, we vary the percentage of the training data used from 5% to 20%.
- Each configuration is repeated 10 times.
Wine Quality Classification (Cont’d)

Experiments

Y. Zhang & D.-Y. Yeung (CSE, HKUST)
Handwritten Letter Classification

- The handwritten letter classification application consists of seven tasks.
- Each task is a binary letter classification problem.
  - The corresponding letters for each task are: c/e, g/y, m/n, a/g, a/o, f/t and h/n.
- For each task, there are about 1000 positive and 1000 negative data points.
Experiments

Handwritten Letter Classification (Cont’d)

Accuracy

ITML
RDML
LDML
TML

# of training examples in each class

Y. Zhang & D.-Y. Yeung (CSE, HKUST)
Experiments

USPS Digit Classification

- There are nine classification tasks.
  - Each task corresponding to the classification of two successive digits.
- The experimental settings are the same as those for handwritten letter classification.
USPS Digit Classification (Cont’d)
Outline

1. Introduction
2. Multi-Task Metric Learning
3. Transfer Metric Learning by Learning Task Relationships
4. Experiments
5. Conclusion
We have proposed a transfer metric learning method to alleviate the labeled data deficiency problem in the target learning task by exploiting useful information from some source tasks. The learning of the distance metric in the target task and the relationships between the source tasks and the target task is formulated as a convex optimization problem.

Future work:
- We will extend our method to semi-supervised setting by exploiting useful information contained in the unlabeled data as well.
Thank you for your attention!