A Theoretical Analysis of Metric Hypothesis Transfer Learning

Michaël Perrot\textsuperscript{1} and Amaury Habrard\textsuperscript{1}
\{michael.perrot,amaury.habrard\}@univ-st-etienne.fr

\textsuperscript{1}Université de Lyon, Université Jean Monnet de Saint-Etienne, Laboratoire Hubert Curien, CNRS, UMR5516, F-42000, Saint-Etienne, France.
1 Introduction

2 General Analysis

3 Specific Loss Analysis and Experiments

4 Conclusion
Metric Learning

Learning how to compare objects: learn a new space where some constraints are fulfilled, e.g. move closer circles of the same color (class) and keep far away circles of different colors (classes).

**Mahalanobis-like Distance**

\[ d_M(x, x') = \sqrt{(x - x')^T M (x - x')}, \quad M \text{ a PSD matrix (} M = LL^T \). \]

**Well-known distances**

- Euclidean Distance: \( M = I \)
- Original Mahalanobis Distance: \( M = \Sigma^{-1} \)
- Zero Distance: \( M = 0 \)
Regularized Metric Learning

\[
\arg \min_{M \geq 0} L_T(M) + \lambda \|M\|_F^2
\]

(1)

with:

- \( T = \{z_i = (x_i, y_i)\}_{i=1}^n \subset (\mathcal{X} \times \mathcal{Y})^n \), a learning sample
- \( L_T(M) = \frac{1}{n^2} \sum_{z, z'} \in T l(M, z, z') \)

with \( l(M, z, z') \):
  - convex with respect to \( M \)
  - \((\sigma, m)\)-admissible
  - \( k\)-lipschitz
  - penalizing high distances between similar examples et small distances between dissimilar examples
- \( \| \cdot \|_F \), the Frobenius norm
Regularized Metric Learning

\[
\arg \min_{M \succeq 0} L_T(M) + \lambda \|M - 0\|_F^2
\]  \hspace{1cm} (1)

with:

- \( T = \{z_i = (x_i, y_i)\}_{i=1}^n \subset (\mathcal{X} \times \mathcal{Y})^n \), a learning sample
- \( L_T(M) = \frac{1}{n^2} \sum_{z, z' \in T} l(M, z, z') \)
  with \( l(M, z, z') \):
    - convex with respect to \( M \)
    - \((\sigma, m)\)-admissible
    - \( k\)-lipschitz
    - penalizing high distances between similar examples et small distances between dissimilar examples
- \( \| \cdot \|_F \), the Frobenius norm

- \( M \), a fixed metric biasing the regularization, e.g. \( I, \Sigma^{-1}, \) a metric learned from another domain, ...

Objective:
Provide a theoretical analysis of biased regularized metric learning and propose an efficient way to reweight the source metric.
Biased Regularized Metric Learning

\[
\arg \min_{M \succeq 0} L_T(M) + \lambda \|M - M_S\|_F^2
\]  

with:

- \(T = \{z_i = (x_i, y_i)\}_{i=1}^n \subset (\mathcal{X} \times \mathcal{Y})^n\), a learning sample
- \(L_T(M) = \frac{1}{n^2} \sum_{z, z' \in T} l(M, z, z')\) with \(l(M, z, z')\):
  - convex with respect to \(M\)
  - \((\sigma, m)\)-admissible
  - \(k\)-lipschitz
  - penalizing high distances between similar examples and small distances between dissimilar examples
- \(\| \cdot \|_F\), the Frobenius norm
- \(M_S\), a fixed metric biasing the regularization, e.g., \(I, \Sigma^{-1}\), a metric learned from another domain, ...
Biased Regularized Metric Learning

\[ \arg \min_{M \succeq 0} L_T(M) + \lambda \| M - M_S \|_F^2 \]  

(1)

with:

- \( T = \{ z_i = (x_i, y_i) \}_{i=1}^n \subset (\mathcal{X} \times \mathcal{Y})^n \), a learning sample
- \( L_T(M) = \frac{1}{n^2} \sum_{z, z' \in T} l(M, z, z') \)
  with \( l(M, z, z') \):
    - convex with respect to \( M \)
    - \((\sigma, m)\)-admissible
    - \( k \)-lipschitz
    - penalizing high distances between similar examples and small distances between dissimilar examples
- \( \| \cdot \|_F \), the Frobenius norm
- \( M_S \), a fixed metric biasing the regularization,
  e.g. \( \mathbf{I}, \Sigma^{-1} \), a metric learned from another domain, ... 

Hypothesis Transfer Learning has already been studied in a different setting [Kuzborskij and Orabona, 2013, 2014].
Biased Regularized Metric Learning

\[
\arg\min_{M \succeq 0} L_T(M) + \lambda \| M - M_S \|_F^2
\]

with:
- \( T = \{ z_i = (x_i, y_i) \}_{i=1}^n \subset (\mathcal{X} \times \mathcal{Y})^n \), a learning sample
- \( L_T(M) = \frac{1}{n^2} \sum_{z, z' \in T} l(M, z, z') \)
  - convex with respect to \( M \)
  - \((\sigma, m)\)-admissible
  - \(k\)-lipschitz
  - penalizing high distances between similar examples et small distances between dissimilar examples
- \( \| \cdot \|_F \), the Frobenius norm
- \( M_S \), a fixed metric biasing the regularization, e.g. \( I, \Sigma^{-1} \), a metric learned from another domain, ...

**Objective:** Provide a theoretical analysis of biased regularized metric learning and propose an efficient way to reweight the source metric.
General Definitions

$(\sigma, m)$-admissibility

A loss function is $(\sigma, m)$-admissible for metric learning if the loss difference between two pairs of examples is bounded by a constant $\sigma$ times a quantity only related to the labels plus a constant:

$$|l(M, z_1, z_2) - l(M, z_3, z_4)| \leq \sigma |y_1 y_2 - y_3 y_4| + m.$$ 

$k$-lipschitz continuity

A loss function is $k$-lipschitz continuous if the loss difference between two metrics is bounded by a constant $k$ times a quantity which only depends on the difference between the two metrics:

$$|l(M, z, z') - l(M', z, z')| \leq k \|M - M'\|_F.$$
Introduction

General Analysis
- On Average Analysis
- Uniform Stability Analysis

Specific Loss Analysis and Experiments

Conclusion

M. Perrot and A. Habrard
On Average Replace Two Stability

The expected loss difference when replacing two examples in the training set is bounded by a value decreasing in $O \left( \frac{1}{n} \right)$.

Extension to metric learning of [Shalev-Shwartz et al., 2010].

**Definition (On-average-replace-two-stability)**

Let $\epsilon : \mathbb{N} \rightarrow \mathbb{R}$ be monotonically decreasing and let $U(n)$ be the uniform distribution over $\{1 \ldots n\}$. A metric learning algorithm is on-average-replace-two-stable with rate $\epsilon(n)$ if for every distribution $\mathcal{D}_T$:

$$
\mathbb{E}_{T \sim \mathcal{D}_T^n \atop i,j \sim U(n) \atop z_1,z_2 \sim \mathcal{D}_T} \left[ l(M_{ij}^*, z^i, z^j) - l(M^*, z^i, z^j) \right] \leq \epsilon(n)
$$

where $M^*$, respectively $M_{ij}^*$, is the optimal solution when learning with the training set $T$, respectively $T_{ij}$. $T_{ij}$ is obtained by replacing $z^i$, the $i^{th}$ example of $T$, by $z_1$ to get a training set $T^i$ and then by replacing $z^j$, the $j^{th}$ example of $T^i$, by $z_2$. 
On Average Bound

The learned metric is on average at least as good as the source metric.

**Theorem (On-average-replace-two-stability)**

Given a training sample $T$ of size $n$ drawn i.i.d. from $\mathcal{D}_T$, an algorithm solving optimization problem (1) is on-average-replace-two-stable with $\epsilon(n) = \frac{8k^2}{\lambda n}$.
On Average Bound

The learned metric is on average at least as good as the source metric.

Theorem (On-average-replace-two-stability)

Given a training sample $T$ of size $n$ drawn i.i.d. from $D_T$, an algorithm solving optimization problem (1) is on-average-replace-two-stable with $\epsilon(n) = \frac{8k^2}{\lambda n}$.

Theorem (On average bound)

For any convex, $k$-lipschitz loss, we have:

$$\mathbb{E}_{T \sim D_T^n} [L_{D_T} (M^*)] \leq L_{D_T} (M_S) + \frac{8k^2}{\lambda n}$$

where the expected value is taken over size-$n$ training sets.
Uniform Stability

Changing an example in the training set does not change much the outcome of the algorithm.

**Definition (Uniform stability [Bousquet and Elisseeff, 2002, Jin et al., 2009])**

An algorithm has a uniform stability in $\epsilon(n)$ if $\forall i$, $\sup_{z, z' \sim D_T} \left| l(M^*, z, z') - l(M^*_i, z, z') \right| \leq \epsilon(n)$

where $M^*$ is the matrix learned on the training set $T$ and $M^*_i$ is the matrix learned on the training set $T^i$ obtained by replacing the $i^{th}$ example of $T$ by a new independent one.
Generalisation Bound

The biased regularized metric learning framework is consistent.

**Theorem (Uniform stability)**

*Given a training sample $T$ of $n$ examples drawn i.i.d. from $D_T$, an algorithm solving optimization problem (1) has a uniform stability in $\epsilon(n) = \frac{4k^2}{\lambda n}$.***
Generalisation Bound

The biased regularized metric learning framework is consistent.

**Theorem (Uniform stability)**

Given a training sample $T$ of $n$ examples drawn i.i.d. from $D_T$, an algorithm solving optimization problem (1) has a uniform stability in
\[ \epsilon(n) = \frac{4k^2}{\lambda n}. \]

**Theorem (Generalization bound)**

With probability $1 - \delta$, for any matrix $M^*$ learned with an $\epsilon(n)$ uniformly stable algorithm and for any convex, $k$-lipschitz and $(\sigma, m)$-admissible loss, we have:
\[ L_{D_T}(M^*) \leq L_T(M^*) + (4\sigma + 2m + c) \sqrt{\frac{\ln \frac{2}{\delta}}{2n}} + O \left( \frac{1}{n} \right) \]

where $c$ is a constant linked to the $k$-lipschitz property of the loss and $\epsilon(n)$ appears in $O \left( \frac{1}{n} \right)$.
3 Specific Loss Analysis and Experiments
Application to a Specific Loss

We consider the following loss (inspired from [Jin et al., 2009]):

\[ l(M, z, z') = \left[ yy'((x - x')^T M (x - x') - \gamma_{yy'}) \right]_+ \] (2)

where \([\cdot]_+\) is the hinge loss, \(yy' = 1\) for examples of the same class and \(-1\) otherwise and \(\gamma_{yy'}\) is the chosen margin.

**Lemma ((\(\sigma, m\))-admissibility)**

Let \(z_1, z_2, z_3, z_4\) be four examples and \(M^*\) be the optimal solution of Problem 1. The convex and \(k\)-lipschitz loss function \(l(M, z, z')\) is \((\sigma, m)\)-admissible with \(\sigma = \max(\gamma_{y_3y_4}, \gamma_{y_1y_2})\) and

\[ m = 2 \max_{x, x'} \|x - x'\|^2 \left( \sqrt{\frac{L_T(M_S)}{\lambda}} + \|M_S\|_F \right). \]
Application to a Specific Loss

We consider the following loss (inspired from [Jin et al., 2009]) :

\[ l(M, z, z') = \left[ yy'((x - x')^T M (x - x') - \gamma_{yy'}) \right]_+ \]  

(2)

where \([\cdot]_+\) is the hinge loss, \(yy' = 1\) for examples of the same class and \(-1\) otherwise and \(\gamma_{yy'}\) is the chosen margin.

Theorem (Generalization bound)

With probability \(1 - \delta\) for any matrix \(M^*\) learned by an algorithm solving optimization problem (1) with loss (2), we have :

\[ L_{DT}(M^*) \leq L_T(M^*) + 4 \left( \sqrt{\frac{L_T(M_S)}{\lambda}} + \|M_S\|_F + c_\gamma \right) \sqrt{\frac{\ln \frac{2}{\delta}}{2n}} + O \left( \frac{1}{n} \right) \]

where \(c_\gamma\) is a constant linked to the k-lipschitz property of the loss and the chosen margins.
Reweighting the Source Metric

Let $M_S = \beta M_{\text{SOURCE}}$, we want to minimize the right hand side of the bound, i.e. to choose the best matrix to transfer. Hence, we search $\beta$ such that:

$$\beta^* = \arg\min_{\beta} \sqrt{\frac{L_T(\beta M_{\text{SOURCE}})}{\lambda}} + \|\beta M_{\text{SOURCE}}\|_F$$

(3)
Reweighting the Source Metric

Let $M_S = \beta M_{\text{SOURCE}}$, we want to minimize the right hand side of the bound, i.e. to choose the best matrix to transfer. Hence, we search $\beta$ such that:

$$
\beta^* = \arg \min_\beta \sqrt{\frac{L_T(\beta M_{\text{SOURCE}})}{\lambda}} + \|\beta M_{\text{SOURCE}}\|_F
$$

(3)

Interest of Tuning $\beta$

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Baselines</th>
<th>Solving optimization problem (1) with loss (2)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1-NN</td>
<td>ITML</td>
</tr>
<tr>
<td>Breast</td>
<td>95.31 ± 1.11</td>
<td>95.40 ± 1.37</td>
</tr>
<tr>
<td>Pima</td>
<td>67.92 ± 1.95</td>
<td>68.13 ± 1.86</td>
</tr>
<tr>
<td>Scale</td>
<td>78.73 ± 1.69</td>
<td>87.31 ± 2.35</td>
</tr>
<tr>
<td>Wine</td>
<td>93.40 ± 2.70</td>
<td>93.82 ± 2.63</td>
</tr>
</tbody>
</table>
Application to a Transfer Learning Task

Setting

The idea is to learn a metric on a source domain and to use this metric to bias the regularizer when learning on the target domain.

MHTL : Metric Hypothesis Transfer Learning
Application to a Transfer Learning Task

Setting

The idea is to learn a metric on a source domain and to use this metric to bias the regularizer when learning on the target domain.

On the Office-Caltech dataset

<table>
<thead>
<tr>
<th>Task</th>
<th>1-NN&lt;sub&gt;S&lt;/sub&gt;</th>
<th>MMDT</th>
<th>GFK</th>
<th>(M_S = \beta \Sigma^{-1})</th>
<th>(M_S = \beta M_{\text{ITML}})</th>
<th>(M_S = \beta M_{\text{LMNN}})</th>
</tr>
</thead>
<tbody>
<tr>
<td>A → C</td>
<td>35.95 ± 1.30</td>
<td><strong>39.76 ± 2.25</strong></td>
<td>37.81 ± 1.85</td>
<td>32.65 ± 3.76</td>
<td>32.93 ± 4.60</td>
<td>34.66 ± 3.66</td>
</tr>
<tr>
<td>A → D</td>
<td>33.58 ± 4.37</td>
<td>54.25 ± 4.32</td>
<td>51.54 ± 3.55</td>
<td>54.69 ± 3.96</td>
<td>51.54 ± 4.03</td>
<td><strong>54.72 ± 5.00</strong></td>
</tr>
<tr>
<td>A → W</td>
<td>33.68 ± 3.60</td>
<td>64.91 ± 5.71</td>
<td>59.36 ± 4.30</td>
<td>67.11 ± 5.11</td>
<td>64.09 ± 5.20</td>
<td><strong>67.62 ± 5.18</strong></td>
</tr>
<tr>
<td>C → A</td>
<td>37.37 ± 2.95</td>
<td><strong>51.05 ± 3.38</strong></td>
<td>46.36 ± 2.94</td>
<td>50.15 ± 4.87</td>
<td>49.89 ± 5.25</td>
<td>50.36 ± 4.67</td>
</tr>
<tr>
<td>C → D</td>
<td>31.89 ± 5.77</td>
<td>52.80 ± 4.84</td>
<td><strong>58.07 ± 3.90</strong></td>
<td>56.77 ± 4.63</td>
<td>53.78 ± 7.23</td>
<td>57.44 ± 4.48</td>
</tr>
<tr>
<td>C → W</td>
<td>28.60 ± 6.13</td>
<td>62.75 ± 5.19</td>
<td>63.26 ± 5.89</td>
<td>64.64 ± 6.44</td>
<td>64.00 ± 6.08</td>
<td><strong>65.11 ± 5.25</strong></td>
</tr>
<tr>
<td>D → A</td>
<td>33.59 ± 1.77</td>
<td><strong>50.39 ± 3.40</strong></td>
<td>40.77 ± 2.55</td>
<td>49.48 ± 4.41</td>
<td>49.11 ± 4.09</td>
<td>49.67 ± 4.00</td>
</tr>
<tr>
<td>D → C</td>
<td>31.16 ± 1.19</td>
<td><strong>35.70 ± 3.25</strong></td>
<td>30.64 ± 1.98</td>
<td>32.90 ± 3.14</td>
<td>32.99 ± 3.58</td>
<td>33.84 ± 2.99</td>
</tr>
<tr>
<td>D → W</td>
<td><strong>76.92 ± 2.18</strong></td>
<td>74.43 ± 3.10</td>
<td>74.98 ± 2.89</td>
<td>65.57 ± 4.52</td>
<td>66.38 ± 6.04</td>
<td>69.72 ± 3.78</td>
</tr>
<tr>
<td>W → A</td>
<td>32.19 ± 3.04</td>
<td>50.56 ± 3.66</td>
<td>43.26 ± 2.34</td>
<td>50.80 ± 3.63</td>
<td>50.16 ± 4.32</td>
<td><strong>50.92 ± 4.00</strong></td>
</tr>
<tr>
<td>W → C</td>
<td>27.67 ± 2.58</td>
<td><strong>34.86 ± 3.62</strong></td>
<td>29.95 ± 3.05</td>
<td>31.54 ± 3.60</td>
<td>31.40 ± 4.29</td>
<td>32.64 ± 3.52</td>
</tr>
<tr>
<td>W → D</td>
<td>64.61 ± 4.30</td>
<td>62.52 ± 4.40</td>
<td><strong>71.93 ± 4.07</strong></td>
<td>57.17 ± 6.50</td>
<td>56.85 ± 5.51</td>
<td>61.14 ± 5.78</td>
</tr>
<tr>
<td>Mean</td>
<td>38.93 ± 3.26</td>
<td><strong>52.83 ± 3.93</strong></td>
<td>50.66 ± 3.28</td>
<td>51.12 ± 4.55</td>
<td>50.26 ± 5.02</td>
<td>52.32 ± 4.36</td>
</tr>
</tbody>
</table>

MHTL, using only the source metric, is competitive with the baselines.
1 Introduction

2 General Analysis

3 Specific Loss Analysis and Experiments

4 Conclusion
Conclusion and Perspectives

We proposed a study of Biased Regularized Metric Learning through:

- An On Average analysis showing that with a fast convergence rate the learned metric is better than the source metric.
- A Consistency Analysis proving that biasing the regularization term toward a source metric does not challenge the consistency of the approach.
- A Reweighting Algorithm allowing us to weight the source metric with respect to the problem at hand when we consider a specific loss.
Conclusion and Perspectives

We proposed a study of Biased Regularized Metric Learning through :

- An On Average analysis showing that with a fast convergence rate the learned metric is better than the source metric.
- A Consistency Analysis proving that biasing the regularization term toward a source metric does not challenge the consistency of the approach.
- A Reweighting Algorithm allowing us to weight the source metric with respect to the problem at hand when we consider a specific loss.

A perspective of this work would be to extend the framework to other settings and other kind of regularizers.


