Mining Social Images with Distance Metric Learning for Automated Image Tagging$^1$

Steven C.H. Hoi
Joint work with Pengcheng Wu, Peilin Zhao, and Ying He

Nanyang Technological University, Singapore

February 11, 2011

$^1$To appear in ACM Int. Conference on Web Search and Data Mining (WSDM2011).
Outline

1. Introduction
2. UDML
   - Problem Setting
   - Generation of Side Information
   - Formulation
   - Algorithm
   - Convergence Analysis
   - Tagging Images with the Optimized Metric
3. Experiments
   - Experimental Testbed
   - Experimental Setup
   - Experimental Results
4. Conclusion
Background

- Web images play an important role in WWW
- Billions of web images do not have proper tags/annotations
- Auto image annotation is the key to make unlabeled photos accessible to web users with existing web search engines
- Regular model-based image annotation by computer vision techniques remains not practical for real applications
- Data-driven image annotation has received more and more attention in recent years
- Due to the great success of social web and social networks, an increasing large amount of images have uploaded to internet and tagged by web users, such as Flickr, Facebook, Picasa, etc.
Background (Cont’)

Figure: Example of a social image from www.Flickr.com

- Can we mine the increasing large amount of (tagged) social images to automatically annotate a new image?
Retrieval based Annotation

- We investigate a data-driven retrieval-based approach for auto image annotation by mining massive social images
- An intuitive example:

![Figure: Example of auto tagging a novel image by our technique](image-url)
## Related Work

Retrieval based Annotation

**Figure:** Diagram for illustrating the process of a retrieval-based annotation paradigm by mining social images with DML

**Open research challenges**

- Learn an **optimal metric** for similarity search
- Build an efficient indexing scheme for fast retrieval
- Rank the tags associated with the top $k$ similar social images
Motivation and Overview

UDML: Unified Distance Metric Learning for Social Image Mining

- Exploit both **visual** and **textual** contents of social images.
  - The optimized metric should induce a small distance for two images of similar visual contents.
  - The optimized metric should induce a small distance for two social images sharing many common textual tags.
- Unify both **inductive** and **transductive** metric learning principles.
Outline

1 Introduction

2 UDML
   - Problem Setting
   - Generation of Side Information
   - Formulation
   - Algorithm
   - Convergence Analysis
   - Tagging Images with the Optimized Metric

3 Experiments
   - Experimental Testbed
   - Experimental Setup
   - Experimental Results

4 Conclusion
Outline

1. Introduction
2. UDML
   - Problem Setting
   - Generation of Side Information
   - Formulation
   - Algorithm
   - Convergence Analysis
   - Tagging Images with the Optimized Metric
3. Experiments
   - Experimental Testbed
   - Experimental Setup
   - Experimental Results
4. Conclusion
Overview of Distance Metric Learning for Mining Social Images

Mahalanobis Distances

Given any two images $x_q \in \mathbb{R}^d$ and $x \in \mathbb{R}^d$, their distance is defined:

$$d_M(x_q, x) = \|x_q - x\|_M^2 = (x_q - x)^\top M(x_q - x)$$

where $M \in \mathbb{R}^{d \times d}$ is any positive semi-definite (PSD) matrix that parameterizes the Mahalanobis distance.

Goal of Distance Metric Learning (DML)

- To learn an optimal matrix $M$ from training data (known as “side information”), aiming to improve the similarity search process towards automated photo tagging

However, side information is only implicitly available in the social image collection.
Outline

1. Introduction

2. UDML
   - Problem Setting
   - Generation of Side Information
   - Formulation
   - Algorithm
   - Convergence Analysis
   - Tagging Images with the Optimized Metric

3. Experiments
   - Experimental Testbed
   - Experimental Setup
   - Experimental Results

4. Conclusion
**Table**: Notations used in the section.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S$</td>
<td>a collection of $N$ social images, i.e. $S = {s_i</td>
</tr>
<tr>
<td>$s_i$</td>
<td>a social image, which consists of two components: visual image and textual tags, i.e., $s_i = (x_i, t_i)$.</td>
</tr>
<tr>
<td>$x_i$</td>
<td>the visual features extracted from the social image.</td>
</tr>
<tr>
<td>$t_i$</td>
<td>the tag vector of the social image.</td>
</tr>
<tr>
<td>$(x, x_+, x_-)$</td>
<td>side information in terms of “triplet” format, where $x$ and $x_+$ are similar/relevant to each other, while $x$ and $x_-$ are dissimilar/irrelevant.</td>
</tr>
<tr>
<td>$R_k(t_{q_i})$</td>
<td>the set of top $k$ social images that are most relevant w.r.t a text-based query $t_{q_i}$.</td>
</tr>
<tr>
<td>$\bar{R}<em>k(t</em>{q_i})$</td>
<td>the set of top $k$ least relevant social images w.r.t $t_{q_i}$.</td>
</tr>
</tbody>
</table>
Generation of Side Information

The Generation Approach

1. Randomly pick a social image from the collection of social images as a query image \( q_i = (x_{q_i}, t_{q_i}) \).

2. Generate a subset of triplets \( P_i \) with respect to \( q_i \) as:

\[
P_i = \{ (x_{q_i}, x_{k_i^+}, x_{k_i^-}) \mid \forall x_{k_i^+} \in R_k(t_{q_i}), \forall x_{k_i^-} \in R_k(t_{q_i}) \}.
\]

3. Repeat the generation process \( N_Q \) times, and form a set of side information \( \{P_i, i = 1 \ldots, N_Q\} \), which will be used as input training data for our distance metric learning task.
Outline

1. Introduction
2. UDML
   - Problem Setting
   - Generation of Side Information
   - Formulation
   - Algorithm
   - Convergence Analysis
   - Tagging Images with the Optimized Metric
3. Experiments
   - Experimental Testbed
   - Experimental Setup
   - Experimental Results
4. Conclusion
Inductive metric learning by maximizing margin

Given side information \( \{ \mathcal{P}_i, i = 1 \ldots, N_Q \} \), we formulate the DML task:

\[
\min_{M \succeq 0} \quad J_1(M) \triangleq \frac{1}{N_p} \sum_{i=1}^{N_Q} \sum_{\forall (x_{q_i}, x_{k_i^+}, x_{k_i^-}) \in \mathcal{P}_i} \ell(M; (x_{q_i}, x_{k_i^+}, x_{k_i^-}))
\]

where \( N_p \) denotes the total number of triplets, and \( \ell \) is a typical hinge loss function defined as:

\[
\ell(M; (x_{q_i}, x_{k_i^+}, x_{k_i^-})) = \max\{0, 1 - [d_M(x_{q_i}, x_{k_i^-}) - d_M(x_{q_i}, x_{k_i^+})]\}
\]

which optimizes the metric by penalizing

1. large distance between two similar images
2. small distance between two dissimilar images
**Transductive fusion of text and visual contents**

Assumption: Two social images of similar tags tend to have similar visual contents. We can formulate this into the following:

$$\min_{M \geq 0} J_2(M) \triangleq \sum_{i,j} w_{ij} \| x_i - x_j \|_M^2$$

where $w_{ij}$ is the cosine similarity between the two textual tag vectors of the two social images, i.e., $w_{ij} = \cos(t_i, t_j)$.

**Remark**

The above formulation indicates that if two social images share similar textual tags, we expect the optimized distance metric induces a small visual distance between them.
UDML Formulation

Transductive fusion of text and visual contents

By decomposing $M$ into a linear mapping $A : \mathbb{R}^d \mapsto \mathbb{R}^r$ where

$$A = [a_1, \ldots, a_r] \in \mathbb{R}^{d \times r}$$

such that $M = AA^\top$, we can rewrite the distance measure as:

$$d_M(x_q, x) = \|x_q - x\|^2_M = (x_q - x)^\top AA^\top (x_q - x)$$

Thus, we can rewrite the previous objective function as:

$$J_2(M) = \sum_{i,j} w_{ij} \|x_i - x_j\|^2_M = \sum_{k=1}^r a_k^\top X(D - W)X^\top a_k$$

$$= \sum_{k=1}^r a_k^\top XLX^\top a_k = tr(XLX^\top M)$$

where $D$ is a diagonal matrix w.r.t. matrix $W$, and $L = D - W$ is known as the Laplacian matrix.
Unified distance metric learning

Combining both inductive and transductive formulations, we have the final formulation of UDML:

$$\min_{M \succeq 0} J(M) \triangleq \frac{1}{2} \text{tr}(M^\top M) + CJ_1(M) + \lambda J_2(M)$$  \hspace{1cm} (1)

where $C > 0$ and $\lambda > 0$ are two trade-off parameters.

Remark

- $\frac{1}{2} \text{tr}(M^\top M)$ is introduced to penalize the norm of the metric to prevent some values of the metric dominating all other elements.

- UDML problem is a convex optimization task, and it belongs to semi-definite programming (SDP),
Outline

1 Introduction

2 UDML
   - Problem Setting
   - Generation of Side Information
   - Formulation
   - Algorithm
   - Convergence Analysis
   - Tagging Images with the Optimized Metric

3 Experiments
   - Experimental Testbed
   - Experimental Setup
   - Experimental Results

4 Conclusion
The stochastic gradient descent algorithm for UDML

1. We randomly sample a subset of triplets from the whole set of triplets:

\[ A_t = \{ (x_{qi}, x_{k_i^+}, x_{k_i^-}) | i \in [Q] \} \]

where \(|A_t| = N_a \ll N_p\).

2. From \(A_t\), we derive an active set of triplets whose values of the loss function are nonzero, i.e.,

\[ A_t^+ = \{ (x_{qi}, x_{k_i^+}, x_{k_i^-}) \in A_t | \ell(M; (x_{qi}, x_{k_i^+}, x_{k_i^-})) > 0 \}. \]
The stochastic gradient descent algorithm for UDML

3 Based on the set of triplets $A_t$, we rewrite the objective function of UDML as follows:

$$J(M; A_t) = \frac{1}{2} \text{tr}(M^\top M) + \lambda \text{tr}(XLX^\top M) + \frac{C}{N_a} \sum_{(x_{q_i}, x_{k_i^+}, x_{k_i^-}) \in A_t} \ell(M; (x_{q_i}, x_{k_i^+}, x_{k_i^-}))$$

4 We solve it by a gradient descent approach, where the sub-gradient is computed as follows:

$$\frac{\partial J(M; A_t)}{\partial M} = M + \lambda XLX^\top - \frac{C}{N_a} \sum_{(x_{q_i}, x_{k_i^+}, x_{k_i^-}) \in A_t^+} [(x_{k_i^-} - x_{q_i})(x_{k_i^-} - x_{q_i})^\top - (x_{k_i^+} - x_{q_i})(x_{k_i^+} - x_{q_i})^\top]$$

5 Repeat the above SGD process until it converges.
Algorithm

**Algorithm 1:** The Stochastic Gradient Descent Algorithm for Unified Distance Metric Learning. (UDML)

**INPUT:** parameter $C$, $\lambda$ and the number of iterations $T$

**PROCEDURE**

1. Choose $M_1$ s.t. $\|M_1\| \leq \sqrt{2C}$
2. for $t = 1, 2, \ldots, T$ do
3. Randomly choose a set $A_t$, s.t. $|A_t| = N_a$
4. Set $A_t^+ = \{(x_{k_i}^+, x_{k_i}^-) \in A_t | \ell(M_t; (x_q, x_{k_i}^+, x_{k_i}^-)) > 0\}$
5. Set a learning rate $\eta_t = \frac{1}{t}$
6. Set $M_{t+1/2} = M_t - \eta_t[\partial J(M_t; A_t)/\partial M]$  
7. Set $M_{t+1} = \min\{1, \frac{\sqrt{2C}}{\|M_{t+1/2}\|_F}\}M_{t+1/2}$
8. end for
9. Project $M_{T+1}^{psd} = PSD(M_{T+1})$

**OUTPUT:** $M_{T+1}^{psd}$

END
Introduction

UDML

1. Problem Setting
2. Generation of Side Information
3. Formulation
4. Algorithm
5. Convergence Analysis
6. Tagging Images with the Optimized Metric

Experiments

1. Experimental Testbed
2. Experimental Setup
3. Experimental Results

Conclusion
Convergence Analysis

Theorem 1

Let us denote by \( M^* \) the optimal solution, and let \( \delta \in (0, 1) \). Assume \( A_t \) is chosen i.i.d from the set of all triplets. Then, with probability of at least \( 1 - \delta \) over the choices of \( A_1, ..., A_T \), we have the bound for the solution of Algorithm 1:

\[
J(M_{T+1}) \leq J(M^*) + \frac{R^2 \ln(T)}{\delta T}
\]

where \( R = \sqrt{2C + (4\lambda + 8C)R_1^2} \) is a constant by assuming \( \|x_j\|_2 \leq R_1 \ \forall j \in [N] \).

Remark

The details of the proof can be found in the paper.
Outline

1. Introduction

2. UDML
   - Problem Setting
   - Generation of Side Information
   - Formulation
   - Algorithm
   - Convergence Analysis
   - Tagging Images with the Optimized Metric

3. Experiments
   - Experimental Testbed
   - Experimental Setup
   - Experimental Results

4. Conclusion
Tagging Images with the Optimized Metric

The process of automated image tagging

1. Retrieve a set of $k$-nearest neighbors of a novel unlabeled image $x_q$, i.e.,

   $$\mathcal{N}_k(x_q) = \{i \in [1, \ldots, n] | x_i \in \text{kNN} - \text{List}(x_q)\},$$

   where $n$ is the total number of images, and the $\text{kNN} - \text{List}$ is found by measuring the distances with the optimized metric $M$, i.e., $\|x_q - x_i\|_M^2$.

2. Define a set of candidate tags $\mathcal{T}_w$ as:

   $$\mathcal{T}_w = \bigcup_{i \in \mathcal{N}_k(x_q)} \mathcal{T}_i$$

   where $\mathcal{T}_i$ represents the set of tags associated with social image $s_i$. 

The process of automated image tagging (Cont’)

3. Calculate the frequency of each candidate tag $w \in \mathcal{T}_w$, denoted as $f(w)$, indicating the total number of times the tag being associated with the set of top $k$ social images.

4. Assign the query image $x_q$ with the tags of high frequency and small average distance, i.e.,

$$w^* = \arg \max_{w \in \mathcal{T}_w \land w \notin \mathcal{T}_q} \frac{f(w)}{\text{avg}_d M(x_q, w) + \kappa}$$

where $\text{avg}_d M(x_q, w)$ represents the average distance with the optimized metric $M$ between the query image and those candidate social images having tag $w$, and $\kappa$ is a smoothing parameter fixed to 1 in experiments.
Outline

1. Introduction
2. UDML
   - Problem Setting
   - Generation of Side Information
   - Formulation
   - Algorithm
   - Convergence Analysis
   - Tagging Images with the Optimized Metric
3. Experiments
   - Experimental Testbed
   - Experimental Setup
   - Experimental Results
4. Conclusion
Experimental Testbed

Outline

1 Introduction
2 UDML
   - Problem Setting
   - Generation of Side Information
   - Formulation
   - Algorithm
   - Convergence Analysis
   - Tagging Images with the Optimized Metric
3 Experiments
   - Experimental Testbed
   - Experimental Setup
   - Experimental Results
4 Conclusion
A real-world social images testbed

- 200,000 images were crawled from Flickr website.
- Each social image’s associated tags were adopted as its textual features.
  - The top 100,000 of most frequent tags were used.
  - Some clearly noisy tags/stopwords were removed.
- Four kinds of global visual features were extracted.
  - Including grid color moment, local binary pattern, Gabor wavelet texture, and edge direction histogram.
  - In total, a 297-dimensional feature vector was used to represent each image.
- 200,000 images were split to 3 sets:
  - The training set: 15,000 images.
  - The test set: 2,000 images.
  - The database set: 183,000 images
Outline

1. Introduction

2. UDML
   - Problem Setting
   - Generation of Side Information
   - Formulation
   - Algorithm
   - Convergence Analysis
   - Tagging Images with the Optimized Metric

3. Experiments
   - Experimental Testbed
   - Experimental Setup
   - Experimental Results

4. Conclusion
Experimental Setup

Settings for the learning process:
- 100 triplets were generated for each query image sampled from the training set.
- The sampling process were repeated 1000 times.
- Tradeoff parameters: $\lambda = 1$, $C = 10000$
- The size of active set: $N_a = 100$
- The total number of iterations: $T = 1000$

Settings for the tagging process:
- For each query image in the test set, top $k$ similar ($k = 30$ in default) images from the database set were retrieved,
- Top $t$ tags were suggested to tag the query image.

Performance evaluation metrics:
- average precision at top $t$ ranked tags
- average recall at top $t$ ranked tags
## Compared Methods

- Baseline (without DML)
- DCA (CVPR’06(Hoi & Lyu, 2006))
- RCA (JMLR’05(Bar-Hillel et al., 2005))
- ITML (ICML’07(Davis et al., 2007))
- RDML (MMSJ’06(Si et al., 2006))
- LMNN (NIPS’05(Weinberger et al., 2006))
- NCA (NIPS’05(J. Goldberger & Salakhutdinov, 2005))
- OASIS (JMLR’10(Chechik et al., 2010))
- LRML (CVPR’08(Hoi, 2008))
- pRCA (MM’09(Wu et al., 2009))
- UDML (the proposed algorithm)
Outline

1 Introduction

2 UDML
   - Problem Setting
   - Generation of Side Information
   - Formulation
   - Algorithm
   - Convergence Analysis
   - Tagging Images with the Optimized Metric

3 Experiments
   - Experimental Testbed
   - Experimental Setup
   - Experimental Results

4 Conclusion
where $k = 30$. 

![Average precision at top $t$ annotated tags](image)
where $k = 30$. 
Experimental Results

The precision-recall curves

$\text{Average Recall}$

$\text{Average Precision}$

where $k = 30.$
Observation

- All DML-based approaches performed better than the baseline.

- The proposed UDML method considerably surpassed all the other approaches for most cases.

- The precision and recall values of all the methods are not high. The possible reasons include:
  - It may be difficult to find similar social images without a very large-scale database;
  - The associated tags of some social image are quite noisy, which could degrade the tagging performance;
  - The extracted image features may be effective enough to describe visual contents of images.
Evaluation under varied top $k$ similar social images

Observation

- If $k$ is too small, some relevant images may not be retrieved.
- If $k$ is too large, lots of irrelevant images can be retrieved, leading to engage many noisy tags in the list of candidate tags.
- Setting $k$ to about 40-50 tend to have the best result.
## Experimental Results

<table>
<thead>
<tr>
<th>Baseline</th>
<th>DCA</th>
<th>RCA</th>
<th>ITML</th>
<th>RDML</th>
<th>LMNN</th>
<th>NCA</th>
<th>OASIS</th>
<th>LRML</th>
<th>pRCA</th>
<th>UDML</th>
</tr>
</thead>
<tbody>
<tr>
<td>nature</td>
<td>nature</td>
<td>nature</td>
<td>nature</td>
<td>nature</td>
<td>nature</td>
<td>nature</td>
<td>nature</td>
<td>nature</td>
<td>nature</td>
<td>nature</td>
</tr>
<tr>
<td>blue</td>
<td>blue</td>
<td>blue</td>
<td>blue</td>
<td>blue</td>
<td>blue</td>
<td>blue</td>
<td>blue</td>
<td>blue</td>
<td>blue</td>
<td>blue</td>
</tr>
<tr>
<td>reflection</td>
<td>reflection</td>
<td>reflection</td>
<td>reflection</td>
<td>reflection</td>
<td>reflection</td>
<td>reflection</td>
<td>reflection</td>
<td>reflection</td>
<td>reflection</td>
<td>reflection</td>
</tr>
<tr>
<td>bird</td>
<td>bird</td>
<td>bird</td>
<td>bird</td>
<td>bird</td>
<td>bird</td>
<td>bird</td>
<td>bird</td>
<td>bird</td>
<td>bird</td>
<td>bird</td>
</tr>
<tr>
<td>animal</td>
<td>animal</td>
<td>animal</td>
<td>animal</td>
<td>animal</td>
<td>animal</td>
<td>animal</td>
<td>animal</td>
<td>animal</td>
<td>animal</td>
<td>animal</td>
</tr>
<tr>
<td>water</td>
<td>water</td>
<td>water</td>
<td>water</td>
<td>water</td>
<td>water</td>
<td>water</td>
<td>water</td>
<td>water</td>
<td>water</td>
<td>water</td>
</tr>
<tr>
<td>canon</td>
<td>canon</td>
<td>canon</td>
<td>canon</td>
<td>canon</td>
<td>canon</td>
<td>canon</td>
<td>canon</td>
<td>canon</td>
<td>canon</td>
<td>canon</td>
</tr>
<tr>
<td>egret</td>
<td>egret</td>
<td>egret</td>
<td>egret</td>
<td>egret</td>
<td>egret</td>
<td>egret</td>
<td>egret</td>
<td>egret</td>
<td>egret</td>
<td>egret</td>
</tr>
<tr>
<td>specanimal</td>
<td>specanimal</td>
<td>specanimal</td>
<td>specanimal</td>
<td>specanimal</td>
<td>specanimal</td>
<td>specanimal</td>
<td>specanimal</td>
<td>specanimal</td>
<td>specanimal</td>
<td>specanimal</td>
</tr>
<tr>
<td>wildlife</td>
<td>wildlife</td>
<td>wildlife</td>
<td>wildlife</td>
<td>wildlife</td>
<td>wildlife</td>
<td>wildlife</td>
<td>wildlife</td>
<td>wildlife</td>
<td>wildlife</td>
<td>wildlife</td>
</tr>
</tbody>
</table>

**Figure:** Examples showing the tagging results by 11 methods.
Outline

1. Introduction

2. UDML
   - Problem Setting
   - Generation of Side Information
   - Formulation
   - Algorithm
   - Convergence Analysis
   - Tagging Images with the Optimized Metric

3. Experiments
   - Experimental Testbed
   - Experimental Setup
   - Experimental Results

4. Conclusion
Conclusion

- We proposed a novel unified distance metric learning (UDML) method for mining social images towards automated photo tagging.
- UDML exploits both textual and visual contents for learning the distance metric in a unified and systematic framework.
- We proposed an efficient stochastic gradient descent algorithm and proved its convergence property theoretically.
- Experimental results on a real social image testbed showed that our UDML method is effective and promising.

Future Work

- Enlarge the social image database;
- Explore other contents of social images;
- Investigate more sophisticated tag ranking techniques.
Thank You!

Contact: Steven Hoi
E-mail: chhoi@ntu.edu.sg
Appendix: Convergence Analysis

**Lemma 1**

The optimal solution of optimization problem (1) is in the convex close set $\mathcal{B}_M = \{ M | \| M \|_F \leq \sqrt{2C} \}$, where $\| \cdot \|_F$ denotes the Frobenius norm.

**Proof**

Denoting by $M^*$ the optimal solution, from the fact that $J(M^*; X) \leq J(0; X)$,

$$ \frac{1}{2} \| M^* \|_F^2 = \frac{1}{2} tr((M^*)^\top M^*) \leq J(M^*; X) \leq J(0; X) = C $$

The second inequality is guaranteed by

$$ tr(XLX^\top M) = \sum_{i,j} w_{ij} \| x_i - x_j \|_M^2 \geq 0 \text{ and } \ell(M; (x_{k_i^+}, x_{k_i^-})) \geq 0 \ . $$
Convergence Analysis

**Lemma 2**

Let $g_1, ..., g_T$ be a sequence of $\sigma$-strongly convex functions w.r.t the function $\frac{1}{2}\| \cdot \|_F^2$. Let $B$ be a closed convex set and define $\Pi_B(M) = \arg\min_{M' \in B} \| M - M' \|_F$. Let $M_1, \ldots, M_{T+1}$ be a sequence of matrices such that $M_1 \in B$ and for $t \geq 1$, $M_{t+1} = \Pi_B(M_t - \eta_t \nabla_t)$, where $\nabla_t$ is a subgradient of $g_t$ at $M_t$ and $\eta_t = 1/(\sigma t)$. Assume that for all $t$, $\| \nabla_t \| \leq G$. Then for all $M \in B$ we have

$$
\frac{1}{T} \sum_{t=1}^{T} g_t(M_t) \leq \frac{1}{T} \sum_{t=1}^{T} g_t(M) + \frac{G^2(1 + \ln(T))}{2\sigma T}
$$

The above lemma is generalized from Hazan2007 (Hazan et al., 2007).
Convergence Analysis

Theorem 1

Assume that \( \|x_{q_i}\| \leq R_1 \ \forall \ i \in [Q] \), \( \|x_j\|_2 \leq R_1 \ \forall j \in [N] \), and \( W \) is normalized such that \( \sum_{i,j} W_{ij} = 1 \). Let \( M^* \) be the optimal solution. Then, for \( T \geq 3 \) we have

\[
\frac{1}{T} \sum_{t=1}^{T} J(M_t; A_t) \leq \frac{1}{T} \sum_{t=1}^{T} J(M^*; A_t) + \frac{R^2 \ln(T)}{T}
\]

where \( R = \sqrt{2C} + (4 \lambda + 8C) R_1^2 \).

Remark

The detailed proof to the above theorem can be found in the paper.
Convergence Analysis

**Theorem 2**

Assume that the conditions stated in Theorem 1 hold and for all $t$, $A_t$ is chosen i.i.d from the set of all triplets. Let $r$ be an integer picked uniformly at random from $[T]$. Then

$$
\mathbb{E}_{A_t} \mathbb{E}_r [J(M_r)] \leq J(M^*) + \frac{R^2 \ln(T)}{T}
$$

**Remark**

The detailed proof to the above theorem can be found in the paper.
Convergence Analysis

Theorem 3

Assume that the conditions stated in Theorem 2 holds. Let $\delta \in (0, 1)$. Then, with probability of at least $1 - \delta$ over the choices of $A_1, \ldots, A_T$ and the index $r$, we have the following bound:

$$J(M_r) \leq J(M^*) + \frac{R^2 \ln(T)}{\delta T}$$

Remark

The detailed proof to the above theorem can be found in the paper.
Convergence Analysis

The convergence of the last matrix $M_{T+1}$

Treat $T + 1$ as a random index drawn from $\{1, \ldots, \hat{T}\}$, where $\hat{T} > T + 1$. Since $M_{T+1}$ does not depend on $M_{T+2}, \ldots, M_{\hat{T}}$, we can terminate the algorithm after $T$ iterations and return $M_{T+1}$. Using Theorem 3, we know that

$$J(M_{T+1}) - J(M^*) \leq \frac{R \ln(\hat{T})}{\delta \hat{T}} \leq \frac{R \ln(T)}{\delta T}$$

where the last inequality holds as $\frac{\ln(T)}{T}$ decreases in $[3, +\infty)$. 


