Unsupervised Riemannian Metric Learning for Histograms Using Aitchison Transformations

Tam Le, Marco Cuturi
Graduate School of Informatics
Kyoto University, Japan

ICML, Lille 2015
Unsupervised Riemannian Metric Learning for Histograms Using Aitchison Transformations

\[ \{x_i, y_i\}_{1 \leq i \leq m} \]

[Xing et al.’02]
[Davis et al.’07]
[Weinberger et al.’06’09]

\[ \{x_i\}_{1 \leq i \leq m} \]

[Lebanon ’06]
[Wang et al.’07]
Unsupervised Riemannian Metric Learning for Histograms Using Aitchison Transformations

\[ f(x) = \sqrt{x} \]

\[ d(x, z) = \arccos(\sqrt{x}^T \sqrt{z}) \]

[Aitchison’86]

[Amari & Nagaoka’00]
Why Focus on Histograms?
Why Focus on Histograms?

\[ P_n = \{ x \in \mathbb{R}^{n+1} \mid x \geq 0 \text{ and } x^T 1 = 1 \} \]
Supervised Metric Learning for Histograms

Kedem et al.’12: Chi-square distance, linear map.

\[ \chi^2(Lx, Lz) \]

\[ L1 = 1, L \geq 0 \]

Cuturi & Avis’14: EMD distance, ground metric

\[ d_M(x, z) = \min_{X1=x, X^T1=z} \langle X, M \rangle \]

\[ X \geq 0 \]

\[ M_{ij} \geq 0, M_{ii} = 0, M_{ij} \leq M_{ik} + M_{kj}, \forall i, j, k \]
Unsupervised Metric Learning for Histograms

Lebanon’06: Pull-back metric, a family of specific transformations

\[ d_\lambda(x, z) = \arccos \left( \sqrt{\frac{x \cdot \lambda}{\langle x, \lambda \rangle}} \sqrt{\frac{z \cdot \lambda}{\langle z, \lambda \rangle}} \right) \]

\[ \lambda \in \text{int}\mathbb{P}_n \]

Reformulation with Aitchison’s perturbation operator

\[ x \oplus \lambda = C(x \cdot \lambda); \quad C(x) = \frac{x}{x^T 1} \]

\[ d_\lambda(x, z) = \arccos \left( \sqrt{x \oplus \lambda}^T \sqrt{z \oplus \lambda} \right) \]
Aitchison Transformation with Fisher Information Metric

\[ f(x) = \sqrt{x} \]

\[ d(u, v) = \arccos(u^T v) \]

\[ d_\gamma(x, z) = \arccos(\sqrt{\gamma(x)^T \gamma(z)}) \]
Aitchison Transformation with Fisher Information Metric

\[ f(x) = \sqrt{x} \]

\[ d(u, v) = \arccos(u^T v) \]

\[ d_\gamma(x, z) = \arccos(\sqrt{\gamma(x)^T \sqrt{\gamma(z)}}) \]
Aitchison Geometry

Perturbation operator

\[ x \oplus z = C(x \bullet z) \in \text{int}\mathbb{P}_n \]

Powering operator

\[ t \circ x = C(x^t) \in \text{int}\mathbb{P}_n \]

where \( x, z \in \text{int}\mathbb{P}_n, t \in \mathbb{R} \)

and \( C(x) = \frac{x}{x^T1} \)

[Aitchison, 1986]
Perturbation Operator

\[ x \oplus \lambda \]

\[ \lambda = [0.3, 0.3, 0.4] \]

\[ \lambda = [0.28, 0.34, 0.38] \]
Powering Operator

$t \odot x$

$t = 0.6$

$t = 2$
Aitchison Transformations

Generalized powering operator

$$\alpha \otimes x = C(x^\alpha) \in \text{int}P_n$$

General Aitchison transformations

$$\gamma : \text{int}P_n \rightarrow \text{int}P_n$$

$$x \mapsto (\alpha \otimes x) \oplus \lambda$$

where $$x, \lambda \in \text{int}P_n, \alpha \in \mathbb{R}_+^{n+1}$$
Generalized Powering Operator

\[ \alpha \otimes x \]

\[ \alpha = [1, 1, 0.5] \]

\[ \alpha = [1.3, 1, 0.5] \]
General Aitchison Transformations

\[(\alpha \otimes \mathbf{x}) \oplus \lambda\]

\[
\begin{align*}
\alpha &= [0.5, 1, 2] \\
\lambda &= [0.2, 0.35, 0.45]
\end{align*}
\]
Aitchison Transformations with Fisher Information Metric

\[
d_{\gamma}(x, z) = \arccos\left(\sqrt{\gamma(x)^T} \sqrt{\gamma(z)}\right)
\]

\[
\gamma(x) = (\alpha \otimes x) \oplus \lambda
\]

\[
\lambda \in \text{int}P_n, \alpha \in \mathbb{R}^{n+1}_+
\]
Maximize Inverse Volume Framework: Pull-back Metric

\[ h = f \circ \gamma \]

\[ f(x) = \sqrt{x} \]

\[ d(u, v) = \arccos(u^T v) \]
Volume element of Riemannian metric $J$ at point $x$:

$$d\text{vol} J(x) = \sqrt{\det G(x)}$$

where $G_{ij} = J(r_i, r_j)$

and $\{r_j\}_{1 \leq j \leq n}$: a basis of $T_x \mathbb{P}_n$
Compute Gram Matrix via Push-forward Map

\[ T_x P_n \]

\[ h : T_x P_n \rightarrow T_{h(x)} S_n^+ \]

\[ \mathbf{r} \mapsto \nabla h(\mathbf{x})|_\mathbf{r} \]

\[ J(\mathbf{r}_i, \mathbf{r}_j) = \langle h_\ast \mathbf{r}_i, h_\ast \mathbf{r}_j \rangle \]
Inverse Volume Element

dvolJ^{−1}(x) \propto \frac{\left(\left(x^\alpha \bullet \lambda\right)^T 1\right)^{\frac{n+1}{2}}}{\left(\left(\frac{x}{\alpha}\right)^T 1\right) g\left(x^{\frac{\alpha-2}{2}}\right)}

where \( g(c) = \prod_k c_k \)
Maximize inverse volume

Volume element summarizes “size” of Riemannian metric.

Inverse volume element measures “smallness” of the metric.

\[ \{x_i\}_{1 \leq i \leq m} \]
Unsupervised Riemannian Metric Learning

\[
\max_{\alpha, \lambda} \quad \mathcal{F} = \frac{1}{m} \sum_{i=1}^{m} \log \frac{\text{dvol} J^{-1}(x_i)}{\int_{\mathbb{P}_n} \text{dvol} J^{-1}(x) \, dx} - \mu \| \log \alpha \|_2^2
\]

s.t. \quad \lambda \in \text{int} \mathbb{P}_n, \quad \alpha \in \mathbb{R}^{n+1}_+

The optimization problem is non-convex.

Maximum pseudo log-likelihood function under the model

\[
p(x) = \frac{\text{dvol} J^{-1}(x)}{\int_{\mathbb{P}_n} \text{dvol} J^{-1}(z) \, dz}
\]
Gradient Ascent

At iteration $t$, we can update for $\alpha, \lambda$

$$\alpha_{t+1} = \Pi \left( \alpha_t + \frac{t_0^\alpha}{\sqrt{t}} \frac{\partial F}{\partial \alpha} \right)$$

$$\lambda_{t+1} = C \left( \lambda_t \cdot \exp \left( \frac{t_0^\lambda}{\sqrt{t}} \frac{\partial F}{\partial \lambda} \right) \right)$$

where $\Pi(\cdot)$ is the projection on $\mathbb{R}_+^{n+1}$ offset by a threshold $\varepsilon = 10^{-20}$
Gradient

\[
\frac{\partial F}{\partial \lambda} = \frac{1}{m} \sum_{i=1}^{m} \frac{\partial \log \text{dvol} J^{-1}(x_i)}{\partial \lambda} - \mathbb{E}_{p(x)} \left( \frac{\partial \log \text{dvol} J^{-1}(x)}{\partial \lambda} \right)
\]

where

\[
\frac{\partial \log \text{dvol} J^{-1}(x)}{\partial \lambda} = \frac{(n + 1)x^\alpha}{2 (x^\alpha \cdot \lambda)^T 1}
\]

Similar for \( \frac{\partial F}{\partial \alpha} \)
Gradient

\[
\frac{\partial F}{\partial \lambda} = \frac{1}{m} \sum_{i=1}^{m} \frac{\partial \log d\text{vol} J^{-1}(x_i)}{\partial \lambda} - E_p(x) \left( \frac{\partial \log d\text{vol} J^{-1}(x)}{\partial \lambda} \right)
\]

where

\[
\frac{\partial \log d\text{vol} J^{-1}(x)}{\partial \lambda} = \frac{(n + 1)x^\alpha}{2 (x^\alpha \cdot \lambda)^T 1}
\]

Similar for \( \frac{\partial F}{\partial \alpha} \)

25
Approximate Gradient by Contrastive Divergence

Approximate $E_p(x)(\cdot)$ by drawing samples from

$$p(x) = \frac{d\text{vol}J^{-1}(x)}{\int_{\mathbb{P}_n} d\text{vol}J^{-1}(z)dz}$$

Use MCMC sampling since only a ratio between probabilities is required.

Metropolis – Hasting sampling method with logistic normal distribution.

[Aitchison & Shen, 1980]
[Blei & Lafferty, 2006]
## Experimental Setting:

### $k$-Medoids Clustering

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Baseline Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIT SCENE</td>
<td>Euclidean distance ($L_2$)</td>
</tr>
<tr>
<td>UIUC SCENE</td>
<td>Total variation distance ($L_1$)</td>
</tr>
<tr>
<td>OXFORD FLOWER</td>
<td>Hellinger distance ($Hellinger$)</td>
</tr>
<tr>
<td>CALTECH-101</td>
<td>Chi-square distance ($Chi_2$)</td>
</tr>
<tr>
<td>20 NEWS GROUP</td>
<td>Cosine similarity ($Cosine$)</td>
</tr>
<tr>
<td>REUTERS</td>
<td>Aitchison map ($ILR$) + Euclidean distance</td>
</tr>
<tr>
<td></td>
<td>Pertubation operation + maximize inverse volume ($pFIM$) [Lebanon, 2006]</td>
</tr>
</tbody>
</table>
Results on $k$-Medoids Clustering

MIT SCENE

UIUC SCENE

<table>
<thead>
<tr>
<th>Measure</th>
<th>CHI2</th>
<th>HEL</th>
<th>COSINE</th>
<th>L2</th>
<th>IRL</th>
<th>pFIM</th>
<th>Our method</th>
</tr>
</thead>
<tbody>
<tr>
<td>F measure</td>
<td>0.55</td>
<td>0.5</td>
<td>0.45</td>
<td>0.4</td>
<td>0.35</td>
<td>0.3</td>
<td>0.25</td>
</tr>
</tbody>
</table>

28
Results on $k$-Medoids Clustering

**OXFORD FLOWER**

<table>
<thead>
<tr>
<th>Measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHI2</td>
</tr>
<tr>
<td>L1</td>
</tr>
<tr>
<td>COSINE</td>
</tr>
<tr>
<td>L2</td>
</tr>
<tr>
<td>IRL</td>
</tr>
<tr>
<td>pFIM</td>
</tr>
<tr>
<td>Our method</td>
</tr>
</tbody>
</table>

**CALTECH 101**

<table>
<thead>
<tr>
<th>Measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHI2</td>
</tr>
<tr>
<td>L1</td>
</tr>
<tr>
<td>COSINE</td>
</tr>
<tr>
<td>L2</td>
</tr>
<tr>
<td>IRL</td>
</tr>
<tr>
<td>pFIM</td>
</tr>
<tr>
<td>Our method</td>
</tr>
</tbody>
</table>
Results on $k$-Medoids Clustering

20 NEWS GROUP

REUTERS
Experimental Setting: \( k \)-NN via Locality Sensitive Hashing

[Charikar, 2002]

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Baseline Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIFAR-10</td>
<td>Euclidean distance (( L_2 ))</td>
</tr>
<tr>
<td></td>
<td>Hellinger distance (\textit{Hellinger})</td>
</tr>
<tr>
<td></td>
<td>Mahalanobis distance (\textit{LMNN})</td>
</tr>
<tr>
<td>MNIST-60K</td>
<td>Hellinger mapping with \textit{LMNN} (\textit{Hellinger-LMNN})</td>
</tr>
<tr>
<td></td>
<td>Pertubation operation + maximize inverse volume (\textit{pFIM})</td>
</tr>
</tbody>
</table>
Results on $k$-NN via LSH

CIFAR–10

Accuracy

Number of bits – $b$

\begin{align*}
\text{CIFAR–10} & \quad 0.4 \\
& \quad 0.35 \\
& \quad 0.34 \\
& \quad 0.33 \\
& \quad 0.32 \\
& \quad 0.31 \\
& \quad 0.3 \\
& \quad 0.29 \\
& \quad 0.28 \\
& \quad 0.27 \\
& \quad 0.26 \\
& \quad 0.25 \\
& \quad 0.24 \\
& \quad 0.23 \\
& \quad 0.22 \\
& \quad 0.21 \\
& \quad 0.2 \\
& \quad 0.19 \\
& \quad 0.18 \\
& \quad 0.17 \\
& \quad 0.16 \\
& \quad 0.15 \\
& \quad 0.14 \\
& \quad 0.13 \\
& \quad 0.12 \\
& \quad 0.11 \\
& \quad 0.1 \\
& \quad 0.09 \\
& \quad 0.08 \\
& \quad 0.07 \\
& \quad 0.06 \\
& \quad 0.05 \\
\end{align*}

$\varepsilon$–value of LHS

\begin{align*}
\text{CIFAR–10} & \quad 2 \\
& \quad 1.8 \\
& \quad 1.6 \\
& \quad 1.4 \\
& \quad 1.2 \\
& \quad 1 \\
& \quad 0.8 \\
& \quad 0.6 \\
& \quad 0.4 \\
& \quad 0.2 \\
\end{align*}

Legend:
- L2
- HELLINGER
- LMNN
- HELLINGER–LMNN
- pFIM
- Our method
Results on $k$-NN via LSH

MNIST–60K

Accuracy vs. Number of bits $b$

Accuracy vs. $\varepsilon$-value of LHS

- L2
- HELLMINGER
- LMNN
- HELLMINGER–LMNN
- pFIM
- Our method
Summary

• Propose a new unsupervised metric learning for histograms that leverages Aitchison transformations.

• Provide a new algorithm to solve a key step for maximizing inverse volume framework by using the contrastive divergence.

• Be able to apply for large datasets via locality sensitive hashing.

• Improve performance of alternative approaches on many benchmark datasets.
Unsupervised Riemannian Metric Learning for Histograms Using Aitchison Transformations

Tam Le, Marco Cuturi
Graduate School of Informatics
Kyoto University, Japan

ICML, Lille 2015
Euclidean Geometry for Simplex?

(Image credit: Cuturi)
Hellinger Geometry for Simplex?

Geometry of the Simplex

$\sqrt{\mathbf{r}}$ is better
Fβ measure

- Precision (P) & Recall (R)

\[ P = \frac{TP}{TP + FP}, \quad R = \frac{TP}{TP + FN}. \]

- Fβ measure:

\[ F_\beta = \frac{(\beta^2 + 1)PR}{\beta^2P + R}. \]

where
- TP: true positive
- TN: true negative
- FP: false positive
- FN: false negative

\[ \beta = \sqrt{\frac{|D|}{|S|}} \]

Penalize FN more strongly than FP
Charikar (2002) proposed a hash function

\[ h_r(\bar{x}) = \text{sign}(r^T \bar{x}) \]

where \( r \) is a random unit-length vector in \( \mathbb{R}^{n+1} \)

\[
\Pr [h_r(\bar{x}) = h_r(\bar{z})] = 1 - \frac{d(x, z)}{\pi}
\]

We use \( b \) hash functions to obtain hash keys (\( b \) hash bits) for each histogram. The complexity to approximate nearest neighbor search is \( O(m^{1/(1+\varepsilon)}) \) where \( m \) is a number of samples.
Locality Sensitive Hashing to Approximate k-NN

- We choose $N = O(m^{1/(1+\varepsilon)})$ random permutation of the bits.

- For each permutation, we maintain a sorted order of the bit vectors.

- Given a query bit, we use a binary search on each permutation to retrieve 2 closest bit vectors.

- We examine $2N$ bit vectors and return $k$ nearest neighbors via Hamming distance to the query bit.
## Experiments: Set up & Parameters

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Samples</th>
<th>#Class</th>
<th>Feature</th>
<th>Rep</th>
<th>#Dim</th>
<th>#Run</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIT Scene</td>
<td>1600</td>
<td>8</td>
<td>SIFT</td>
<td>BoF</td>
<td>200</td>
<td>100</td>
</tr>
<tr>
<td>UIUC Scene</td>
<td>3000</td>
<td>15</td>
<td>SIFT</td>
<td>BoF</td>
<td>200</td>
<td>100</td>
</tr>
<tr>
<td>OXFORD Flower</td>
<td>1360</td>
<td>17</td>
<td>SIFT</td>
<td>BoF</td>
<td>200</td>
<td>100</td>
</tr>
<tr>
<td>CALTECH-101</td>
<td>3060</td>
<td>102</td>
<td>SIFT</td>
<td>BoF</td>
<td>200</td>
<td>100</td>
</tr>
<tr>
<td>20 News Group</td>
<td>10000</td>
<td>20</td>
<td>BoW</td>
<td>LDA</td>
<td>200</td>
<td>100</td>
</tr>
<tr>
<td>Reuters</td>
<td>2500</td>
<td>10</td>
<td>BoW</td>
<td>LDA</td>
<td>200</td>
<td>100</td>
</tr>
<tr>
<td>MNIST-60K</td>
<td>60000</td>
<td>10</td>
<td>Normalized Intensity</td>
<td>784</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>CIFAR-10</td>
<td>60000</td>
<td>10</td>
<td>BoW</td>
<td>SIFT</td>
<td>200</td>
<td>4</td>
</tr>
</tbody>
</table>
Riemannian manifold

• Manifold
  – Is a space that is locally homeomorphic to a Euclidean space. [Lee’02].
  – Each point in the manifold has a neighbourhood that is homeomorphic to a Euclidean space. [Wikipedia]

• Differential manifold (smooth manifold)
  – Is a type of manifold that locally similar enough to a linear space to allow one to do calculus. [Wikipedia]

• Riemannian manifold
  – Is a differential manifold equipped with an inner product in the tangent space. [Lee’02]
  – The family of inner product is called Riemannian metric.
Tangent space

• Tangent space: $T_x M, x \in M$ [Lee’02]
  – Set of directional derivatives at $x$ operating on differential functions $C^\infty(M, \mathbb{R})$
  – Classes of curves having the same velocity vectors at $x$.

• Illustration: tangent space on the sphere

$$T_x S_n = \left\{ v \in \mathbb{R}^{n+1} \mid \sum_{i=1}^{n+1} v_i x_i = 0 \right\}$$
Distance in Riemannian manifold

• Length of the tangent vector $v \in T_xM$:
  $$\|v\| = \sqrt{g_x(v, v)}$$

• Length of curves $\gamma : [a, b] \mapsto M$
  - $\gamma'(t)$ is a tangent vector in the tangent space $T_{\gamma(t)}M$
    (for any $t \in (a, b)$) (a.k.a velocity vector of the curve $\gamma$ at time $t$)
    $$L(\gamma) = \int_a^b \sqrt{g_x(\gamma'(t), \gamma'(t))} \, dt$$

• Distance between: $x, y \in M$
  $$d_g(x, y) = \inf_{\gamma \in \Gamma(x, y)} \int_a^b \sqrt{g_x(\gamma'(t), \gamma'(t))} \, dt$$

where $\Gamma(x, y)$: set of differentiable curves connecting $x$ and $y$. 
Pull-back metric

- Given \((N, h)\) and a diffeomorphism \(f : M \mapsto N\), we define a metric \(f^* h\) on \(M\) called pull-back metric by relation:

\[
(f^* h)_x(u, v) = h_{f(x)}(f^*u, f^*v)
\]
Function $f$ between 2 topological space $(X, T_X)$ and $(Y, T_Y)$ is called a homeomorphism if
- $f$: bijection, continuous
- Inverse function $f^{-1}$: continuous