Journal of Machine Learning Research 1 (2014) xx-xx Submitted 10/11; Revised 08/13; Published 1/14

Ground Metric Learning

Marco Cuturi MCUTURI@I.KYOTO-U.AC.JP

David Avis AVIS@I.KYOTO-U.AC.JP

Graduate School of Informatics, Kyoto University
36-1 Yoshida-Honmachi, Sakyo-ku, Kyoto 606-8501, Japan

Editor: Gert Lanckriet

Abstract

Optimal transport distances have been used for more than a decade in machine learning to compare histograms of features. They have one parameter: the *ground metric*, which can be any metric between the features themselves. As is the case for all parameterized distances, optimal transport distances can only prove useful in practice when this parameter is carefully chosen. To date, the only option available to practitioners to set the ground metric parameter was to rely on *a priori* knowledge of the features, which limited considerably the scope of application of optimal transport distances. We propose to lift this limitation and consider instead algorithms that can learn the ground metric using only a training set of labeled histograms. We call this approach ground metric learning. We formulate the problem of learning the ground metric as the minimization of the difference of two convex polyhedral functions over a convex set of metric matrices. We follow the presentation of our algorithms with promising experimental results which show that this approach is useful both for retrieval and binary/multiclass classification tasks.

Keywords: Optimal Transport Distance, Earth Mover's Distance, Metric Learning, Metric Nearness

1. Introduction

We consider in this paper the problem of learning a distance for normalized histograms. Normalized histograms, namely finite-dimensional vectors with nonnegative coordinates whose sum is equal to 1, arise frequently in natural language processing, computer vision, bioinformatics and more generally areas involving complex datatypes. Objects of interest in such areas are usually simplified and are represented as a bag of smaller features. The occurrence frequencies of each of these features in the considered object can be then represented as a histogram. For instance, the representation of images as histograms of pixel colors, SIFT or GIST features (Lowe 1999, Oliva and Torralba 2001, Douze et al. 2009); texts as bags-of-words or topic allocations (Joachims 2002, Blei et al. 2003, Blei and Lafferty 2009); sequences as n-grams counts (Leslie et al. 2002) and graphs as histograms of subgraphs (Kashima et al. 2003) all follow this principle.

Various distances have been proposed in the statistics and machine learning literatures to compare two histograms (Amari and Nagaoka 2001, Deza and Deza 2009, §14). Our focus is in this paper is on the family of optimal transport distances, which is both well motivated theoretically (Villani 2003, Rachev 1991) and works well empirically (Pele and Werman 2009). Optimal transport distances are particularly popular in computer vision, where, following the influential work of Rubner et al. (1997), they were called *Earth Mover's Distances* (EMD).

Optimal transport distances can be thought of as meta-distances that build upon a *metric on the features* to form a *distance on histograms of features.* Such a metric between features, which is known in the computer vision literature as the *ground metric*[1], is the only parameter of optimal transport distances. In their seminal paper, Rubner et al. (2000) argue that, *"in general, the ground distance can be any distance and will be chosen according to the problem at hand"*. As a consequence, the earth mover's distance has only been applied to histograms of features when a good candidate for the ground metric was available beforehand. We argue that this is problematic in two senses: first, this restriction limits the application of optimal transport distances to problems where such a knowledge exists. Second, even when such an *a priori* knowledge is available, we argue that there cannot be a "universal" ground metric that will be suitable for all learning problems involving histograms on such features. As with all parameters in machine learning algorithms, the ground metric should be selected adaptively using data samples. The goal of this paper is to propose *ground metric learning* algorithms to do so.

This paper is organized as follows: after providing background and a few results on optimal transport distances in Section 2, we propose in Section 3 a criterion to select a ground metric given a training set of labeled histograms. We then show how to obtain a local minimum for that criterion using a projected subgradient descent algorithm in Section 4. We provide a review of other relevant distances and metric learning techniques in Section 5, in particular Mahalanobis metric learning techniques (Xing et al. 2003, Weinberger et al. 2006, Weinberger and Saul 2009, Davis et al. 2007) which have inspired much of this work. We provide empirical evidence in Section 6 that the metric learning framework proposed in this paper compares favorably to competing tools in terms of retrieval and classification performance. We conclude this paper in Section 7 by providing a few research avenues that could alleviate the heavy computational price tag of these techniques.

Notations: We consider throughout this paper histograms of length $d \geq 1$. We use upper case letters $A, B, \ldots$ for $d \times d$ matrices. Bold upper case letters $\mathbf{A}, \mathbf{B}, \ldots$ stand for larger matrices; lower case letters $r, c, \ldots$ are used for vectors of $\mathbb{R}^d$ or simply scalars in $\mathbb{R}$. An upper case letter M and its bold lower case $\mathbf{m}$ stand for the same matrix written in $d \times d$ matrix form or d^2 vector form by stacking successively all its column vectors from the left-most on the top to the right-most at the bottom. The notations $\overline{\mathbf{m}}$ and $\underline{\mathbf{m}}$ stand respectively for the strict upper and lower triangular parts of M expressed as vectors of size $\binom{d}{2}$. The order in which these elements are enumerated must be coherent in the sense that the upper triangular part of M^T expressed as a vector must be equal to $\underline{\mathbf{m}}$. Finally, we use the Frobenius dot-product for both matrix and vector representations, written as $\langle A, B \rangle \overset{\text{def}}{=} \operatorname{tr}(A^T B) = \mathbf{a}^T \mathbf{b}$.

[1] Since the terms *metric* and *distance* are interchangeable mathematically speaking, we will always use the term *metric* for a metric between features and the term *distance* for the resulting transport distance between histograms, or more generally any other distance on histograms.

2. Optimal Transport Between Histograms

We recall in this section a few facts about optimal transport between two histograms. A more general and technical introduction is provided by Villani (2003, Introduction and §7); practical insights and motivation for the application of optimal transport distances in machine learning can be found in Rubner et al. (2000); a recent review of extensions and acceleration techniques to compute the EMD can be found in (Pele and Werman 2009, §2).

Our interest in this paper lies in defining distances for pairs of probability vectors, namely on two nonnegative vectors r and c with the same sum. We consider in the following vectors of length d, and define the probability simplex accordingly:

$$\Sigma_d \overset{\text{def}}{=} \{u \in \mathbb{R}_+^d \mid \sum_{i=1}^{d} u_i = 1\}.$$

Optimal transport distances build upon two ingredients: (1) a $d \times d$ metric matrix, known as the ground metric parameter of the distance; (2) a feasible set of $d \times d$ matrices known as the transport polytope. We provide first an intuitive description of optimal transport distances in Section 2.1 (which can be skipped by readers familiar with these concepts) and follow with a more rigorous exposition in Section 2.2.

2.1 The Intuition behind Optimal Transport

The fundamental idea behind optimal transport distances is that they can be used to compare histograms of features, when the features lie in a metric space and can therefore be compared one with the other. To illustrate this idea, suppose we wish to compare images of $10 \times 10 = 100$ pixels. Suppose further, for the sake of simplicity, that these pixels can only take values in a range of 4 possible colors, **dark red**, light red, **dark blue** and light blue, and that each image is represented as a histogram of 4 colors as in Figure 1.

So called *bin-to-bin* distances (we provide a formal definition in Section 5.1) would compute the distance between a and b by comparing for each given index i their coordinates a_i and b_i one at a time. For instance, computing the Manhattan distances (the l_1 norm of the difference of two vectors) of three histograms a, b and c in Figure 1, we obtain that a is equidistant to b and c. However, upon closer inspection, assuming that dark and light red have more in common than, say, dark red and dark blue, one may have the intuition that c should be closer to a than it is to b. Optimal transport theory implements this intuition by carrying out an optimization procedure to compute a distance between histograms. Such an optimization procedure builds upon a set of feasible solutions (transport mappings) and a cost function (a linear cost), to define an optimal transport.

Mapping a to b: An assignment between a and b assigns to each of the 100 colored pixels of a one of the 100 colored pixels of b. By grouping these assignments according to the 4×4 possible color pairs, we obtain a 4×4 matrix which details, for each possible pair of colors (i, j), the overall amount x_{ij} of pixels of color i in a which have been morphed into pixels of color j in b. Because such a matrix representation only provides *aggregated*

$$X = \begin{array}{cc} \begin{array}{cccc} \mathbf{13} & \mathbf{7} & \mathbf{56} & \mathbf{24} \end{array} & \\ \left[\begin{array}{cccc} 10 & 4 & 40 & 6 \\ 1 & 1 & 11 & 7 \\ 1 & 2 & 4 & 7 \\ 1 & 0 & 1 & 4 \end{array}\right] & \begin{array}{c} \mathbf{60} \\ \mathbf{20} \\ \mathbf{14} \\ \mathbf{6} \end{array} \end{array} \tag{1}$$

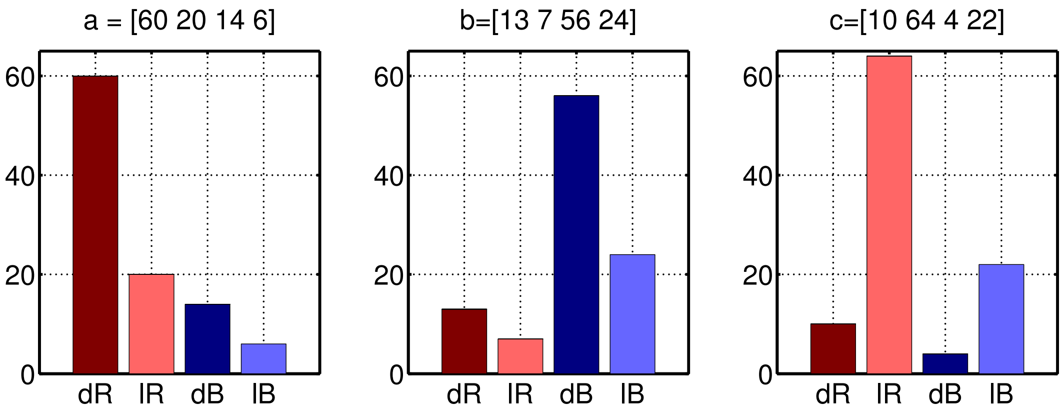

Figure 1: Three color histograms summing to 100. Although a and c are arguably closer to each other because of their overlapping dominance in red colors, the Manhattan distance cannot consider such an overlap and treats all colors separately. As a result, in this example, a is equidistant from b and c, $\|a-b\|_1 = \|a-c\|_1 = 120$.

assignments and does not detail the actual individual assignments, such matrices are known as transport plans. A transport plan between a and b must be such that its row and column sums match the quantities detailed in a and b, as highlighted on the top and right side of an example matrix X in Equation 1.

A Linear Cost for Transport Plans: A cost matrix M quantifies all 16 possible costs m_{ij} of turning a pixel of a given color i into another color j. In the example provided in Equation 3, M states for instance that the cost of turning a dark red pixel into a dark blue pixel is twice that of turning it into a light red pixel; that transferring a colored pixel from a to the *same* color in b has a zero cost for all four colors. The cost of a transport plan X, given the cost matrix M, is defined as the Frobenius dot-product of X and M, namely $\langle X, M \rangle = \sum_{ij} x_{ij} m_{ij} = 169$ in our example.

$$M = \begin{array}{ccccc} \bullet & \bullet & \bullet & \bullet & \\ 0 & 1 & 2 & 3 & \bullet \\ 1 & 0 & 3 & 2 & \bullet \\ 2 & 3 & 0 & 1 & \bullet \\ 3 & 2 & 1 & 0 & \bullet \end{array} \tag{2}$$

Smallest Possible Total Transport Cost: The transport distance is defined as the lowest cost one could possibly find by considering *all* possible transport plans from a to b. Computing such an optimum involves solving a linear program, as detailed in Section 2.3. For a and b and given M above, solving this program would return an optimal matrix $X^\star$ provided in Equation (3) with an optimum of $\langle X^\star, M \rangle = 120$. When comparing a and c, the distance would, on the other hand, be equal to 72. Comparing these two numbers, we can see that the transport distance agrees with our initial intuition that a is closer to c than b by taking into account a metric on features. We define rigorously the properties of both the cost matrix M and the set of transport plans in the next section.

$$X^\star = \begin{array}{ccccc} \mathbf{13} & \mathbf{7} & \mathbf{56} & \mathbf{24} & \\ 13 & & 42 & 5 & \mathbf{60} \\ & 7 & & 13 & \mathbf{20} \\ & & 14 & & \mathbf{14} \\ & & & 6 & \mathbf{6} \end{array} \tag{3}$$

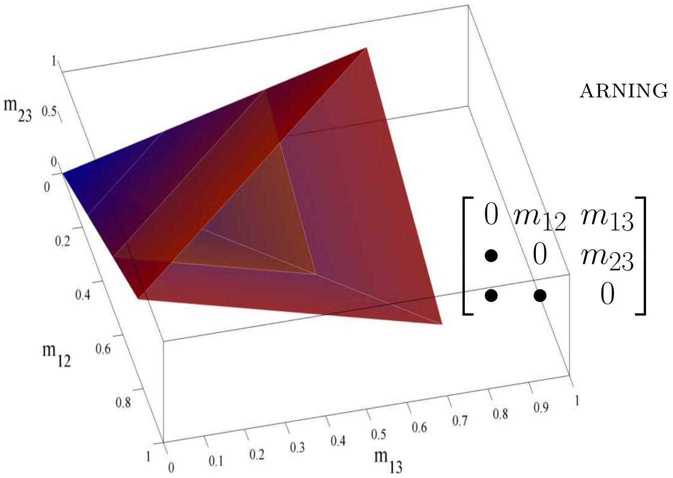

Figure 2: Semimetric cone in 3 dimensions. A $d \times d$ metric matrix for $d = 3$ can be described by 3 positive numbers m_{12}, m_{13} and m_{23} that follow the three triangle inequalities, $m_{12} \leq m_{13} + m_{23}$, $m_{13} \leq m_{12} + m_{23}$, $m_{23} \leq m_{12} + m_{13}$. The set (neither open nor closed) of *positive* triplets (x, y, z) forms a set of metric matrices.

2.2 The Ingredients of Discrete Optimal Transport

Optimal transport distances between histograms are computed through a mathematical program. The feasible set of that program is a polytope of matrices. Its objective is a linear function parameterized by metric matrices. We define both in the sections below.

2.2.1 Objective: Semimetric and Metric Matrices

Consider d points labeled as $\{1, 2, \ldots, d\}$ in a metric space. Form now the $d \times d$ matrix M where element m_{ij} is equal to the distance between points i and j. Because of the metric axioms, the elements of M must obey three rules: (1) symmetry: $m_{ij} = m_{ji}$ for all pairs of indices i, j; (2) $m_{ii} = 0$ for all indices i and more generally $m_{ij} \geq 0$ for any pair (i, j); (3) triangle inequality: $m_{ij} \leq m_{ik} + m_{kj}$, for all triplets of indices i, j, k. The set of all $d \times d$ matrices that observe such rules, and thus represent hypothetically the pairwise distances between d points taken in any arbitrary metric space, is known as the cone of semimetric matrices,

$$\mathcal{M} \overset{\text{def}}{=} \left\{ M \in \mathbb{R}^{d \times d} : \forall\, 1 \leq i, j, k \leq d, m_{ii} = 0, m_{ij} \leq m_{ik} + m_{kj} \right\} \subset \mathbb{R}_+^{d \times d}.$$

Note that the $\binom{d}{2}$ symmetry conditions $m_{ij} = m_{ji}$ and non-negativity conditions $m_{ij} \geq 0$ are contained in the d^3 linear inequalities described in the definition above. $\mathcal{M}$ is a polyhedral set, because it is defined by a finite set of linear equalities and inequalities. $\mathcal{M}$ is also a convex pointed cone as can be visualized in Figure 2 for $d = 3$. Additionally, if a matrix M satisfies conditions (1) and (3) but also has, in addition to (2), the property that $m_{ij} > 0$ whenever $i \neq j$, then we call M a metric matrix. We write $\mathcal{M}_+ \subset \mathcal{M}$ for the set of metric matrices, which is neither open nor closed.

2.2.2 Feasible Set: Transport Polytopes

Consider two vectors r and c in the simplex Σ_d. Let $U(r, c)$ be the set of $d \times d$ nonnegative matrices such that their row and columns sums are equal to r and c respectively, that is, writing $\mathbb{1}_d \in \mathbb{R}^d$ for the column vector of ones,

$$U(r, c) = \{X \in \mathbb{R}_+^{d \times d} \mid X\mathbb{1}_d = r,\ X^\top \mathbb{1}_d = c\}.$$

Because of these constraints, it is easy to see that any matrix $X = [x_{ij}]$ in $U(r, c)$ is such that $\sum_{ij} x_{ij} = 1$. While r and c can be interpreted as two probability measures on the discrete set $\{1, \dots, d\}$, any matrix X in $U(r, c)$ is thus a probability measure on $\{1, \dots, d\} \times \{1, \dots, d\}$, the cartesian product of $\{1, \dots, d\}$ with itself. $U(r, c)$ can be identified with the set of all discrete probabilities on $\{1, \dots, d\} \times \{1, \dots, d\}$ that admit r and c as their first and second marginals respectively.

$U(r, c)$ is a bounded polyhedron (the entries of any X in $U(r, c)$ are bounded between 0 and 1) and is thus a polytope with a finite set of extreme points. This polytope has an effective dimension of $d^2 - 2d + 1$ in the general case where r and c have positive coordinates (Brualdi 2006, §8.1). $U(r, c)$ is known in the operations research literature as the set of transport plans between r and c (Rachev and Rüschendorf 1998). When r and c are integer valued histograms with the same total sum, a transport plan with integral values is also known as a contingency table or a two-way table with fixed margins (Lauritzen 1982, Diaconis and Efron 1985).

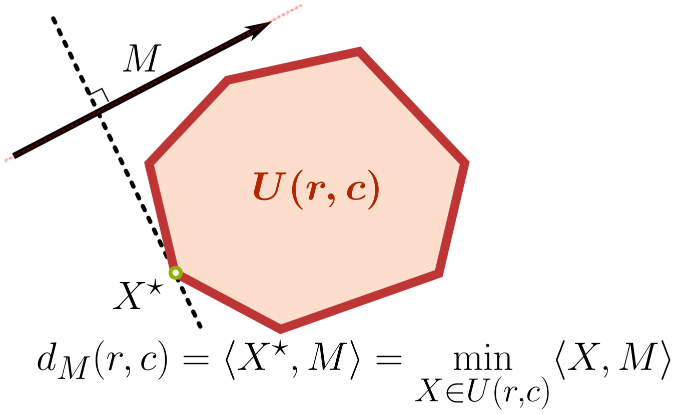
M
$U(r,c)$
$X^\star$
$d_M(r,c) = \langle X^\star, M\rangle = \min_{X \in U(r,c)} \langle X, M\rangle$

Figure 3: Schematic view of the optimal transport distance. Given a feasible set $U(r, c)$ and a cost parameter $M \in \mathcal{M}_+$, the distance between r and c is the minimum of $\langle X, M \rangle$ when X varies across $U(r, c)$. The minimum is reached here at $X^\star$.

2.3 Optimal Transport Distances

Given two histograms r and c of Σ_d and a matrix M, the quantity

$$G(r, c; M) \overset{\text{def}}{=} \min_{X \in U(r,c)} \langle M, X \rangle.$$

describes the optimum of a linear program whose feasible set is defined by r and c and whose cost is parameterized by M. G is a positive homogeneous function of M, that is $G(r, c; tM) = tG(r, c; M)$ for $t \geq 0$. $G(r, c; M)$ can also be described as minus the support function (Rockafellar 1970, §13) of the polytope $U(r, c)$ evaluated at $-M$. A schematic view of that LP is given in Figure 3.

When M belongs to the cone of metric matrices $\mathcal{M}$, the value of $G(r, c; M)$ is a distance (Villani 2003, §7, p.207) between r and c, parameterized by M. In that case, assuming implicitly that M is fixed and only r and c vary, we will refer to $G(r, c; M)$ as $d_M(r, c)$, the optimal transport distance between r and c.

Theorem 1 *d_M is a distance on Σ_d whenever $M \in \mathcal{M}_+$.*

The fact that $d_M(r, c)$ is a distance is a well known result; a standard proof for continuous probability densities is provided in (Villani 2003, Theorem 7.3). A proof often reported in the literature for the discrete case can be found in (Rubner et al. 2000). We believe this proof is not very clear, so we provide an alternative proof in the Appendix.

When r and c are, on the contrary, considered fixed, we will use the notation $G_{rc}(M)$ to stress that M is the variable argument of G, as will be mostly the case in this paper. Although using two notations for the same mathematical object may seem cumbersome, these notations will allow us to stress alternatively which of the three variables r, c and M are considered fixed in our analysis.

2.3.1 Extensions of Optimal Transport Distances

The distance d_M bears many names: 1-Wasserstein; Monge-Kantorovich; Mallow's (Mallows 1972, Levina and Bickel 2001) and finally Earth Mover's (Rubner et al. 2000) in the computer vision literature. Rubner et al. (2000) and more recently Pele and Werman (2009) have also proposed to extend the optimal transport distance to compare unnormalized histograms, that is vectors with nonnegative coordinates which do not necessarily sum to 1. Simply put, these extensions compute a distance between two unnormalized histograms u and v by combining any difference in the total mass of u and v with the optimal transport plan that can carry the whole mass of u onto v if $\|u\|_1 \leq \|v\|_1$ or v onto u if $\|v\|_1 \leq \|u\|_1$. These extensions can also be traced back to earlier work by Kantorovich and Rubinshtein (1958), see Vershik (2006) for a historical perspective. We will not consider such extensions in this work, and will only consider distances for histograms of equal sum.

2.3.2 Relationship with Other Distances

The optimal transport distance bears an interesting relationship with the total variation distance, which is a popular distance between histograms of features in computer vision following early work by Swain and Ballard (1991). As noted by (Villani 2003, p.7 & Ex.1.17 p.36), the total variation distance, defined as

$$d_{\mathrm{TV}}(r, c) \overset{\mathrm{def}}{=} \frac{1}{2}\|r - c\|_1,$$

can be seen as a trivial instance of optimal transport distances by simply noting that

$$d_{\mathrm{TV}} = d_{M_{\mathbb{1}}},$$

where $M_{\mathbb{1}}$ is the matrix of ones with a zero diagonal, namely $M_{\mathbb{1}}(i, j)$ is equal to 1 if $i = j$ and zero otherwise. The metric on features defined by $M_{\mathbb{1}}$ simply states that all d considered features are equally different, that is their pairwise distances are constant. This relationship between total variation and optimal transport can be compared to the analogous observation that Euclidean distances are a trivial instance of the Mahalanobis family of distances, by setting the Mahalanobis parameter to the identity matrix. Tuning the ground metric M to select an optimal transport distance d_M can thus be compared to the idea of tuning a positive-definite matrix Ω to define a suitable Mahalanobis distance for a given problem: Mahalanobis distances are to the Euclidean distance what optimal transport distances are to the total variation distance, as schematized in Figure 4. We discuss this parallel further when reviewing related work in Section 5.2.

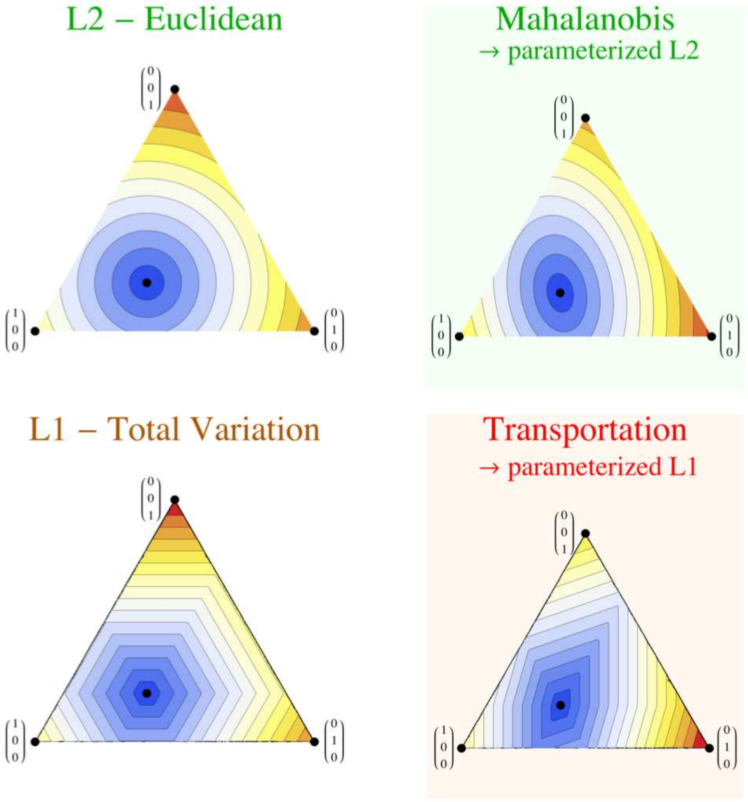

Figure 4: Contour plots of the Euclidean (top-left) and Total variation (bottom-left) of all points in the simplex for $d = 3$ to the point $[0.5, 0.3, 0.2]$, and their respective parameterized equivalents, the Mahalanobis distance (top-right) and the transport distance (bottom-right). The parameter for the Mahalanobis distance has been drawn randomly. The upper right values of the ground metric M are 0.8 and 0.4 on the first row and 0.6 on the second row.

2.3.3 Computing Optimal Transport Distances

The distance d_M between two histograms r and c can be computed as the solution of the following Linear Program (LP),

$$\begin{array}{rll} d_M(r,c) = & \text{minimize} & \sum_{i,j=1}^d m_{ij}x_{ij} \\ & \text{subject to} & \sum_{j=1}^d x_{ij} = r_i, 1 \leq i \leq d \\ & & \sum_{i=1}^d x_{ij} = c_j, 1 \leq j \leq d \\ & & x_{ij} \geq 0, 1 \leq i,j \leq d. \end{array}$$

This program is equivalent to the following program, provided in a more compact form, as:

$$\begin{array}{rll} d_M(r,c) = & \text{minimize} & \mathbf{m}^T\mathbf{x} \\ & \text{subject to} & \mathbf{A}\mathbf{x} = \begin{bmatrix} r \\ c \end{bmatrix}_* \\ & & \mathbf{x} \geq 0, \end{array} \tag{4}$$

where $\mathbf{A}$ is the $(2d-1) \times d^2$ matrix that encodes the row-sum and column-sum constraints for X to be in $U(r,c)$ as

$$\mathbf{A} = \begin{bmatrix} \mathbb{1}_{1\times d} \otimes I_d \\ I_d \otimes \mathbb{1}_{1\times d} \end{bmatrix}_*,$$

where $\otimes$ is Kronecker's product and the lower subscript $\left[\cdot\right]_*$ in a matrix (resp. a vector) means that its last line (resp. element) has been removed. This modification is carried out to make sure that all constraints described by $\mathbf{A}$ are independent, or equivalently that $\mathbf{A}^T$ is not rank deficient. This LP can be solved using the network simplex (Ford and Fulkerson 1962) or through more specialized minimum-cost network flow algorithms (Ahuja et al. 1993, §9). The computational effort required to compute a single distance between two histograms of dimension d scales typically as $O(d^3 \log(d))$ (Pele and Werman 2009, §2.3) when M has no particular structure.

2.4 Properties of the Optimal Transport Distance Seen As a Function of M

When both its arguments are fixed, the optimal transport distance $d_M(r,c)$ seen as a function G_{rc} of M has three important properties: G_{rc} is piecewise linear; concave; a subgradient of G_{rc} can be directly recovered by considering any optimal solution of the linear program considered to compute G_{rc}. These properties are crucial, because they highlight that for a given pair of histograms (r,c), a gradient direction to increase or decrease $d_M(r,c)$ can be obtained through the optimal transport plan that realizes $d_M(r,c)$, and that maximizing this value is a convex problem.

2.4.1 Concavity and Piecewise-linearity

Because its feasible set $U(r,c)$ is a bounded polytope and its objective is linear, Problem (4) has an optimal solution in the finite set $\mathrm{Ex}(r,c)$ of extreme points of $U(r,c)$ (Bertsimas and Tsitsiklis 1997, Theorem 2.7, p.65). G_{rc} is thus the minimum of a finite collection of linear functions, each indexed by an extreme point, and thus

$$G_{rc}(M) = \min_{X\in U(r,c)} \langle X, M\,\rangle = \min_{X\in \mathrm{Ex}(r,c)} \langle X, M\,\rangle, \tag{5}$$

is piecewise linear. G_{rc} is also concave by a standard result stating that the point-wise minimum of a family of affine functions is itself concave (Boyd and Vandenberghe 2004, §3.2.3).

2.4.2 Differentiability

Because the computation of G_{rc} involves a linear program, the gradient ∇G_{rc} of G_{rc} at a given point M is equal to the optimal solution $X^\star$ to Problem (4) whenever this solution is unique,

$$\nabla G_{rc} = X^\star,$$

as stated by Bertsimas and Tsitsiklis (1997, Theorem 5.3). Intuitively, by continuity of all functions involved in Problem (4) and the uniqueness of the optimal solution $X^\star$, one can show that there exists a ball with a positive radius around M for which $G_{rc}(M)$ is locally linear, equal to $\langle X^\star, M\,\rangle$ on that ball, resulting in the fact that the gradient of $\langle X^\star, M\,\rangle$ is simply $X^\star$. More generally and regardless of the uniqueness of $X^\star$, *any* optimal solution $X^\star$ of Problem (4) is in the sub-differential $\partial G_{rc}(M)$ of G_{rc} at M (Bertsimas and Tsitsiklis 1997, Lemma 11.4). Indeed, suppose that $Z(p)$ is the minimum of a linear program Z parameterized by a cost vector x, over a bounded feasible polytope with extreme points $\{c_1, \ldots, c_m\}$. $Z(x)$ can in that case be written as

$$Z(x) = \min_{i=1,\ldots,m} u_i + c_i^T x.$$

Then, defining $E(x) = \{i | Z(x) = u_i + c_i^T x\}$, namely the set of indices of extreme points which are optimal for x, Bertsimas and Tsitsiklis (1997, Lemma 11.4) show that for any fixed x and any index i in $E(x)$, c_i is a subgradient of Z at x. More generally, this lemma also shows that the differential of Z at x is exactly the convex hull of those optimal solutions $\{c_i\}_{i \in E(x)}$. If, as in Equation (5), these c_i's describe the set of extreme points of $U(r, c)$, the variable x is the ground metric M, and Z is G_{rc}, this lemma implies that any optimal transport is necessarily in the subdifferential of $G_{rc}(M)$, and that this subdifferential is exactly the convex hull of all the optimal transports between r and c using cost M.

In summary, the distance $d_M(r, c)$ seen as a function of M ($G_{rc}(M)$ using our notations) can be computed by solving a network flow problem, and any optimal solution of that network flow is a subgradient of the distance with respect to M. This function itself is concave in M. We use extensively these properties in Section 4 when we optimize the criteria considered in the next section.

3. Learning Ground Metrics as an Optimization Problem

We define in this section a family of criteria to quantify the relevance of a ground metric to compare histograms in a given learning task. We use to that effect a training sample of histograms with additional information.

3.1 Training Set: Histograms and Side Information

Suppose that we are given a sample $\{r_1, \ldots, r_n\} \subset \Sigma_d$ of histograms in the canonical simplex along with a family of coefficients $\{\omega_{ij}\}_{1 \leq i,j \leq n}$, which quantify how similar r_i and r_j are. We assume that these coefficients are such that ω_{ij} is positive whenever r_i and r_j describe similar objects and negative for dissimilar objects. We further assume that this similarity is symmetric, $\omega_{ij} = \omega_{ji}$. The similarity of an object with itself will not be considered in the following, so we simply assume that $\omega_{ii} = 0$ for $1 \leq i \leq n$.

In the most simple case, these weights may reflect a labeling of all histograms into multiple classes and be set to $\omega_{ij} > 0$ whenever r_i and r_j come from the same class and $\omega_{ij} < 0$ for two different classes. An ever simpler setting which we consider in our experiments is that of setting $\omega_{ij} = \mathbb{1}_{y_i=y_j}$, where the label y_i of histogram r_i for $1 \leq i \leq n$ is taken in a finite set of labels $\mathcal{L} = \{1, 2, \ldots, L\}$. Let us introduce more notations before moving on to the next section. Since by symmetry $\omega_{ij} = \omega_{ji}$ and $G_{r_i r_j} = G_{r_j r_i}$, we restrict the set of pairs of indices (i, j) we will study to

$$\mathcal{I} \overset{\text{def}}{=} \{(i, j) \mid i, j \in \{1, \ldots, n\}, i < j\},$$

and introduce two subsets of $\mathcal{I}$, the subsets of similar and dissimilar histograms:

$$\mathcal{E}_+ \overset{\text{def}}{=} \{(i, j) \in \mathcal{I} \mid \omega_{ij} > 0\}; \quad \mathcal{E}_- \overset{\text{def}}{=} \{(i, j) \in \mathcal{I} \mid \omega_{ij} < 0\}.$$

Finally, we define the shorthand $G_{ij} \overset{\text{def}}{=} G_{r_i r_j}$.

3.2 Feasible Set of Metrics

We propose to formulate the ground metric learning problem as that of finding a metric matrix $M \in \mathcal{M}_+$ such that the corresponding optimal transport distance d_M computed between pairs of points in $(r_1, \ldots, r_n)$ agrees with the weights ω. However, because projectors are not well defined on feasible sets that are not closed, we will consider the whole of the semimetric cone $\mathcal{M}$ as a feasible set instead of considering $\mathcal{M}_+$ directly. We implicitly assume in this paper that, if our algorithms output a matrix that has null off-diagonal elements, such a matrix will be regularized by adding the same arbitrarily small positive constant to all its off-diagonal elements. Moreover, and as remarked earlier, two histograms r and c define a homogeneous function G_{rc} of M, that is $G_{rc}(tM) = t\, G_{rc}(M)$. To remove this ambiguity on the scale of M, we only consider in the following matrices that lie in the intersection of $\mathcal{M}$ and the unit sphere in $\mathbb{R}^{d\times d}$ of the 1-norm,

$$\mathcal{M}_1 = \mathcal{M} \cap B_1,$$

where $B_1 = \{A \in \mathbb{R}^{d\times d} \mid \|A\|_1 \overset{\text{def}}{=} \|\mathbf{a}\|_1 = 1\}$. $\mathcal{M}_1$ is convex as the intersection of two convex sets. In what follows we call matrices in $\mathcal{M}_1$ metric matrices (this is a slight abuse of language since some of these matrices are in fact semimetrics).

3.3 A Local Criterion to Select the Ground Metric

More precisely, this criterion will favor metrics M for which the distance $d_M(r_i, r_j)$ is *small* for pairs of *similar* histograms r_i and r_j ($\omega_{ij} > 0$) and *large* for pairs of *dissimilar* histograms ($\omega_{ij} < 0$). We build such a criterion by considering the family of all $\binom{n}{2}$ pairs

$$\{\, (\omega_{ij}, G_{ij}(M))\, , (i, j) \in \mathcal{I}\}.$$

Given the i^{th} datum of the training set, we consider the subsets $\mathcal{E}_{i+}$ and $\mathcal{E}_{i-}$ of points that share their label with r_i and those that do not respectively:

$$\mathcal{E}_{i+} \overset{\text{def}}{=} \{j|(i, j) \text{ or } (j, i) \in \mathcal{E}_+\}, \quad \mathcal{E}_{i-} \overset{\text{def}}{=} \{j|(i, j) \text{ or } (j, i) \in \mathcal{E}_-\}.$$

Within these subsets, we consider the sets N_{ik}^+ and N_{ik}^-, which stand for the indices of any k nearest neighbours of r_i using distance d_M and whose indices are taken respectively in the subsets $\mathcal{E}_{i+}$ and $\mathcal{E}_{i-}$. For each index i and corresponding histogram r_i, we can now form the weighted sum of distances to its *similar* and *dissimilar* neighbors

$$S_{ik}^+(M) \overset{\text{def}}{=} \sum_{j \in N_{ik}^+} \omega_{ij}\, G_{ij}(M), \quad \text{and} \quad S_{ik}^-(M) \overset{\text{def}}{=} \sum_{j \in N_{ik}^-} \omega_{ij}\, G_{ij}(M). \tag{6}$$

Note that N_{ik}^+ and N_{ik}^- are not necessarily uniquely defined. Whenever more than one list of indices can qualify as the k closest neighbors of r_i, we select such a list randomly among all possible choices. We adopt the convention that $N_{ik}^+ = \mathcal{E}_{i+}$ whenever k is larger than the cardinality of $\mathcal{E}_{i+}$, and follow the same convention for N_{ik}^-. We use these two terms to form our final criterion:

$$C_k(M) \overset{\text{def}}{=} \sum_{i=1}^{n} S_{ik}^+(M) + S_{ik}^-(M). \tag{7}$$

4. Approximate Minimization of C_k

Since all functions G_{ij} are concave, C_k can be cast as a difference of convex functions

$$C_k(M) = S_k^-(M) - \text{-}S_k^+(M),$$

where both

$$S_k^-(M) \overset{\text{def}}{=} \sum_{i=1}^{n} S_{ik}^-(M) \text{ and } \text{-}S_k^+(M) \overset{\text{def}}{=} \sum_{i=1}^{n} \text{-}S_{ik}^+(M)$$

are convex, by virtue of the convexity of each of the terms S_{ik}^- and -S_{ik}^+ defined in Equation (6). This follows in turn from the concavity of each of the distances G_{ij} as discussed in Sections 2.4 and 3.3, and the fact that such functions are weighted by negative factors, ω_{ij} for $(i, j) \in \mathcal{E}_-$ and -ω_{ij} for $(i, j) \in \mathcal{E}_+$. We propose an algorithm to approximate the minimization of C_k defined in Equation (7) that takes advantage of this decomposition.

4.1 Subdifferentiability of C_k

It is easy to see that, using the results on G_{rc} we have recalled in Section 2.4.1, the gradient of C_k computed at a given metric matrix M is

$$\nabla C_k(M) = \nabla S_k^-(M) + \nabla S_k^+(M),$$

where,

$$\nabla S_k^+(M) = \sum_{i=1}^{n} \sum_{j \in N_{ik}^+} \omega_{ij} X_{ij}^\star, \quad \nabla S_k^-(M) = \sum_{i=1}^{n} \sum_{j \in N_{ik}^-} \omega_{ij} X_{ij}^\star,$$

whenever all solutions $X_{ij}^\star$ to the linear programs G_{ij} considered in C_k are unique and whenever each of the two sets of k nearest neighbors of each histogram r_i is unique. Also as recalled in Section 2.4.1, any optimal solution $X_{ij}^\star$ is in the sub-differential $\partial G_{ij}(M)$ of G_{ij} at M and we thus have that

$$\sum_{i=1}^{n}\sum_{j\in N_{ik}^{+}}\omega_{ij}X_{ij}^{\star}\in\partial S_k^{+}(M),\quad \sum_{i=1}^{n}\sum_{j\in N_{ik}^{-}}\omega_{ij}X_{ij}^{\star}\in\partial S_k^{-}(M),$$

regardless of the unicity of the nearest-neighbors sets of each histogram r_i. The details of the computation of $S_k^-(M)$ and of the subgradient described above are given in Algorithm 1. The computations for $S_k^+(M)$ are analogous to those of $S_k^-(M)$ and we use the abbreviation $S_k^{\pm}(M)$ to consider either of these two cases in our algorithm outline.

Algorithm 1 Computation of $z = S_k^{\pm}(M)$ and a subgradient γ, where ± is either + or −.

```
**Input**: $M \in \mathcal{M}_1$.
**for** $(i,j) \in \mathcal{E}_\pm$ **do**
  Compute the optimum $z_{ij}^\star$ and an optimal solution $X_{ij}^\star$ for Problem (4) with cost vector
  $\mathbf{m}$ and constraint vector $[r_i; r_j]_*$.
**end for**
Set $G = 0, z = 0$.
**for** $i \in \{1,\cdots,n\}$ **do**
  Select the smallest $k$ elements of $z_{ij}^\star, j \in \mathcal{E}_{i\pm}$ to define the set of neighbors $N_{ik}^\pm$.
  **for** $j \in N_{ik}^\pm$ **do**
    $G \leftarrow G + \omega_{ij}X_{ij}^\star$.
    $z \leftarrow z + \omega_{ij}z^\star$.
  **end for**
**end for**
**Output** $z$ and $\gamma = \overline{\mathbf{g}} + \underline{\mathbf{g}}$.
```

4.2 Local Linearization of the Concave Part of C_k

We describe in Algorithm 2 a simple approach to obtain an approximate solution to the problem of minimizing C_k with a projected subgradient descent and a local linearization of the concave part of C_k. Algorithm 2 runs a subgradient descent on C_k using two nested loops: we linearize the concave part of C_k in an outer loop and minimize the resulting convex approximation in the inner loop.

More precisely, the first loop is parameterized with an iteration counter p and starts by computing both S_k^+ (the concave part of C_k) and a vector γ_+ in its subdifferential using the current candidate metric M_p. Using this value and the subgradient γ_+, the concave part S_k^+ of C_k can be locally approximated by its first order Taylor expansion,

$$C_k(M) \approx S_k^-(M) + S_k^+(M_p) + \gamma_+^T(M - M_p).$$

This approximation is convex, larger than C_k and can be minimized in an inner loop using a projected subgradient descent. When this convex function has been minimized up to sufficient precision, we obtain a point

$$M_{p+1} \in \underset{M\in\mathcal{M}_1}{\operatorname{argmin}}\, S_k^-(M) + S_k^+(M_p) + \gamma_+^T(M - M_p).$$

We increment p and repeat the linearization step described above. The algorithm terminates when sufficient progress in the outer loop has been realized, at which point the matrix computed in the last iteration is returned as the output of the algorithm.

The overall quality of the solution obtained through this procedure is directly linked to the quality of the initial point M_0. The selection of M_0 requires thus some attention. We provide a few options to select M_0 in the next section.

Algorithm 2 Projected Subgradient Descent to minimize C_k

```
**Input** $M_0 \in \mathcal{M}_1$ (see Section 4.3), gradient step $t_0$.
$t \leftarrow 1$.
$p \leftarrow 0, \quad M_0^{\text{out}} \leftarrow M_0$.
**while** $p < p_{\max}$ or insufficient progress for $z_p^{\text{out}}$ **do**
  Use Algorithm 1 to compute $z_+ \overset{\text{def}}{=} S_k^+(M_p^{\text{out}})$ and $\gamma_+$.
  $q \leftarrow 0, \quad M_0^{\text{in}} \leftarrow M_p^{\text{out}}$.
  **while** $q < q_{\max}$ or insufficient progress for $z_q^{\text{in}}$ **do**
    Compute $\gamma_-$ and $z_-$ of $S_k^-$ using Algorithm 1 with $M_q^{\text{in}}$, $(i,j) \in \mathcal{E}^-$.
    Set $z_q^{\text{in}} \leftarrow z_- + z_+ + \gamma_+^T(\underline{\mathbf{m}}_q^{\text{in}} - \underline{\mathbf{m}}_p^{\text{out}})$ .
    Set $M_{q+1}^{\text{in}} \leftarrow P_{\mathcal{M}_1}\left(\underline{\mathbf{m}}_q^{\text{in}} - \frac{t_0}{\sqrt{q}}(\gamma_+ + \gamma_-)\right)$.
    $q \leftarrow q + 1$.
    $t \leftarrow t + 1$.
  **end while**
  $M_{p+1}^{\text{out}} \leftarrow M_q^{\text{in}}$.
  $p \leftarrow p + 1$.
**end while**
**Output** $M_p^{\text{out}}$.
```

4.3 Initial Points

Since C_k is not a convex criterion, particular care needs to be taken to initialize our descent algorithm. We propose in this section two approaches to choose the initial point M_0.

4.3.1 The Total Variation Distance as an Optimal Transport Distance

The total variation distance between two histograms, defined as half the l_1 norm of their difference, can provide an educated guess to define an initial point M_0 to optimize C_k. Indeed, as explained in Section 2.3, the total variation distance can be interpreted as the optimal transport distance parameterized with the uniform ground metric $M_{\mathbb{1}}$ which is a matrix equal to 1 on all its off-diagonal terms and 0 on the diagonal. Therefore, we consider $M_{\mathbb{1}}$ (divided by $d(d-1)$ to normalize it) in our experiments to initialize Algorithm 2. Since C_k is not convex, using $M_{\mathbb{1}}$ is attractive from a numerical point of view because $M_{\mathbb{1}}$ exhibits the highest entropy among all matrices in $\mathcal{M}_1$. This choice has, however, two drawbacks:

- Because all the costs enumerated in $M_{\mathbb{1}}$ are equal, one can show that for a pair of histograms (r, c) *any* transport matrix that assigns the maximum weight to its diagonal elements, namely any matrix X in the convex set

$$\{X \in U(r,c) |\, x_{ii} = \min(r_i, c_i)\}$$

is optimal. As a result, any matrix in that set is in the subdifferential of G_{rc} at $M_{\mathbb{1}}$. Solvers that build upon the network simplex will return an arbitrary vertex within that set, mostly depending on the pivot rule they use. The very first subgradient descent iteration is thus likely to be extremely uninformative, and this should be reflected by a poor initial behaviour which we do indeed observe in practice.

- Because such a starting point ignores the information provided by all histograms $\{r_i, 1 \leq i \leq n\}$ and weights $\{\omega_{ij}, (i,j) \in \mathcal{I}\}$, we expect it to be far from the actual optimum.

We propose an alternative approach in the next section: we approximate C_k by a linear function of M and set M_0 to be the minimizer of that approximation.

4.3.2 Linear Approximations to C_k and Independence Tables

We propose to form an initial point M_0 by replacing the optimization underlying the computation of each distance $G_{ij}(M)$ by a dot product,

$$G_{ij}(M) = \min_{X \in U(r_i, r_j)} \langle M, X \rangle \approx \langle M, \Xi_{ij} \rangle,$$

where Ξ_{ij} is a representative matrix of the polytope $U(r_i, r_j)$. This idea is illustrated in Figure 5. We discuss a natural choice to define Ξ_{ij} later in this section. Assuming we have chosen such matrices, we replace now each term G_{ij} in the criterion presented in Equation (7) by the corresponding quantity $\langle M, \Xi_{ij} \rangle$ and obtain an approximation χ_k of C_k parameterized by a matrix Ξ_k,

$$\chi_k(M) \overset{\text{def}}{=} \langle M, \Xi_k \rangle, \text{ where } \Xi_k \overset{\text{def}}{=} \sum_{i=1}^{n} \sum_{j \in N_{ik}^- \cup N_{ik}^+} \omega_{ij}\, \Xi_{ij},$$

where the k nearest neighbors of each histogram r_i defined in N_{ik}^- and N_{ik}^+ are those selected by considering the total variation distance. To select a candidate matrix M that minimizes this criterion, we consider the following penalized problem,

$$\min_{M \in \mathcal{M}} \lambda \langle M, \Xi_k \rangle + \|M\|_2^2 = \min_{M \in \mathcal{M}} \|M + \frac{\lambda}{2} \Xi_k\|_2^2, \quad \lambda > 0, \tag{8}$$

which can be solved using the approach described by Brickell et al. (2008, Algorithm 3.1). Brickell et al. propose *triangle fixing* algorithms to obtain projections on the cone of distances under various norms, including the Euclidean distance. They study in particular the following problem,

$$\min_{M \in \mathcal{M}} \|M - H\|_2, \tag{9}$$

where H is a *symmetric nonnegative* matrix that is *zero on the diagonal.* It is however straightforward to check that these three conditions, although intuitive when considering the metric nearness problem (Brickell et al. 2008, §2), are not necessary for Algorithm (3.1) described by Brickell et al. (2008, §3) to work. This algorithm is not only valid for non-symmetric matrices H as pointed out by the authors themselves, but it is also applicable to matrices H with negative entries and non-zero diagonal entries. Problem (8) can thus be solved by replacing H by $-\frac{\lambda}{2}\Xi_k$ in Problem (9) regardless of the sign of the entries of Ξ.

Note that other approaches could be considered to minimize the dot product $\langle M, \Xi \rangle$ using alternative regularizers. Frangioni et al. (2005) propose for instance to handle linear programs in the intersection between the cone of metrics and the set of polyhedral constraints $\{M_{ik} + M_{kj} + M_{ij} \leq 2\}$ which defines what is known as the metric polytope.

The techniques presented above build upon a linear approximation of each function $G_{ij}(M)$ as $\langle M, \Xi_{ij} \rangle$ by selecting a particular matrix Ξ_{ij} such that $G_{ij}(M) \approx \langle M, \Xi_{ij} \rangle$. We propose to use a simple proxy for the optimal transport distance: the dot-product of M with a matrix that lies at the center of $U(r,c)$, as illustrated in Figure 5. We consider for

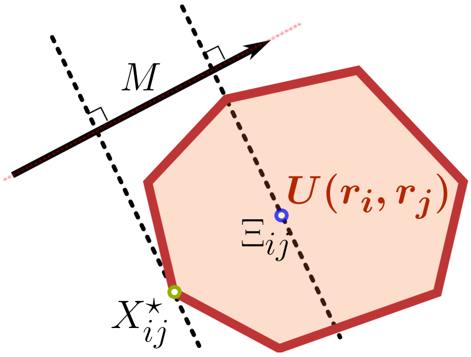

Figure 5: Schematic view of the approximation $\min_{X \in U(r_i,r_j)} \langle M, X \rangle \approx \langle M, \Xi_{ij} \rangle$ carried out when using a central transport table Ξ_{ij} instead of the optimal table $X^\star_{ij}$ to compare r_i and r_j.

such a center the *independence* table rc^T (Good 1963). The table rc^T, which is in $U(r,c)$ because $rc^T \mathbb{1}_d = r$ and $cr^T \mathbb{1}_d = c$, is also the maximal entropy table in $U(r,c)$, that is, the table which maximizes

$$h(X) = -\sum_{p,q=1}^{d} X_{pq} \log X_{pq}.$$

Using the independence table to approximate G_{ij}, that is using the approximation

$$\min_{X \in U(r_i,r_j)} \langle M, X \rangle \approx r_i^T M r_j,$$

provides us with a weighted center,

$$\Xi_k = \sum_{i=1}^{n} \sum_{j \in N_{ik}^- \cup N_{ik}^+} \omega_{ij} r_i r_j^T.$$

Note however that this approximation tends to overestimate substantially the distance between two similar histograms. Indeed, it is easy to check that $r^T M r$ is positive whenever r has positive entropy. In the case where all coordinates of r are equal to $1/d$, $r^T M r$ is $\|M\|_1/d^2$. To close this section, one may notice that several methods can be used to compute centers for polytopes such as $U(r, c)$, among which the Chebyshev center, the analytic center, or the center of the Löwner-John ellipsoid, all described by Boyd and Vandenberghe (2004, §8.4,§8.5). We have not considered these approaches because computing them involve, unlike the independence table proposed above, the resolution of large convex programs or LP's. Barvinok has, on the other hand, proposed recently a new center tailored specifically for transport polytopes, that he calls the typical table (2010). The typical table can be computed efficiently, both in theory and practice, as the result of a convex program of $2d$ variables (Barvinok 2010, p.523). Experimental results indicate that they perform very similarly to independent tables so we do not explore them further in this paper.

In summary, we propose in this section to approximate C_k by a linear function and compute its minimum in the intersection $\mathcal{M}_1$ of the l_1 unit sphere and the cone of metric matrices. This linear objective can be efficiently minimized using a set of tools proposed by Brickell et al. (2008) adapted to our problem. In order to propose such an approximation, we have used the *independence* tables as representative points of the polytopes $U(r_i, r_j)$. The successive steps of the computations that yield an initial point M_0 are given in Algorithm 3.

Algorithm 3 Initial Point M_0 to minimize C_k

```
Set $\Xi = 0$.
for $i \in \{1, \cdots, n\}$ do
    Compute the neighborhood sets $N_{ik}^+$ and $N_{ik}^-$ of histogram $r_i$ using an arbitrary distance, e.g. the total variation distance.
    for $j \in N_{ik}^+ \cup N_{ik}^-$ do
        $\Xi \leftarrow \Xi + \omega_{ij} r_i r_j^T$.
    end for
end for
Set $M_0 \leftarrow \min_{M \in \mathcal{M}} \|M + \frac{\lambda}{2}\Xi\|_2$ (Brickell et al. 2008, Algorithm 3.1).
Output $M_0$. optional: regularize $M_0$ by setting $M_0 \leftarrow \lambda M_0 + (1-\lambda) M_{\mathbb{1}}$.
```

5. Related Work

We provide in this section an overview of other distances for histograms of features. We start by presenting simple distances on histograms and follow by presenting metric learning approaches.

5.1 Metrics on the Probability Simplex

Deza and Deza (2009, §14) provide an exhaustive list of metrics for probability measures, most of which apply to probability measures on $\mathbb{R}$ and $\mathbb{R}^d$. When narrowed down to distances for probabilities on unordered discrete sets – the dominant case in machine learning applications – Rubner et al. (2000, §2) propose to split such distances into two families: *bin-to-bin* distances and *cross-bin* distances. Let $r = (r_1, \dots, r_d)^T$ and $c = (c_1, \dots, c_d)^T$ be two histograms in the canonical simplex Σ_d.

Bin-to-bin distances only compare the d couples of bin-counts $(r_i, c_i)_{i=1..d}$ independently to form a distance between r and c. The Jensen-divergence, χ_2, Hellinger, total variation distances and more generally Csizar f-divergences (Amari and Nagaoka 2001, §3.2) all fall within this category. Notice that any of these divergences is known to work usually better for histograms than a straightforward application of the Euclidean distance as shown in our experiments or for instance by Chapelle et al. (1999, Table 4). This can be explained in theory using geometric (Amari and Nagaoka 2001, §3) or statistical arguments (Aitchison and Egozcue 2005).

Bin-to-bin distances are easy to compute and accurate enough to compare histograms when all d features are sufficiently distinct. When, on the contrary, some of these features are known to be similar, either because of statistical co-occurrence (*e.g.* the words `cat` and `kitty`) or through any other form of prior knowledge (*e.g.* pixel colors or amino-acid similarity) then a simple bin-to-bin comparison may not be accurate enough as argued by Rubner et al. (2000, §2.2). In particular, bin-to-bin distances are invariably large when they compare histograms with distinct supports, regardless of the fact that these two supports may in fact describe very similar features.

Cross-bin distances handle this issue by considering all d^2 possible pairs (r_i, c_j) of cross-bin counts to form a distance. The most simple cross-coordinate distance for general vectors in $\mathbb{R}^d$ is arguably the Mahalanobis family of distances,

$$d^\Omega(x, y) = \sqrt{(x - y)^T \Omega (x - y)},$$

where Ω is a positive definite $d \times d$ matrix. The Mahalanobis distance between x and y can be interpreted as the Euclidean distance between Lx and Ly where L is a Cholesky factor of Ω or any square root of Ω. Learning such linear maps L or positive definite matrices Ω directly using labeled information has been the subject of a substantial amount of research in recent years. We briefly review this literature in the following section.

5.2 Mahalanobis Metric Learning

Xing et al. (2003), followed by Weinberger et al. (2006) and Davis et al. (2007) have proposed different algorithms to learn the parameters of a Mahalanobis distance. We refer to recent surveys by Kulis (2012) and Bellet et al. (2013) for more details on these approaches. These techniques define first a criterion and a feasible set of candidate matrices – either a positive semidefinite matrix Ω or a linear map L – to optimize the best parameter that fits best the data at hand. The criteria we propose in Section 3 are modeled along these ideas. Weinberger et al. (2006) were the first to consider criteria that only use nearest neighbors, which inspired in this work the proposal of C_k in Section 3.3.

We would like point out that Mahalanobis metric learning and ground metric learning have very little in common conceptually: Mahalanobis metric learning algorithms learn a $d \times d$ positive semidefinite matrix or a $m \times d$ linear operator L. Ground metric learning learns instead a $d \times d$ metric matrix M. The difference between Mahalanobis distances and optimal transport distances can be further highlighted by these simple identities:

$$d_{\mathrm{TV}}(r,c) = \frac{1}{2}\|r-c\|_1 = d_{M_{\mathbb{1}}}(r,c), \quad d_2(r,c) = \|r-c\|_2 = d^I(r,c)$$

The relationship between the Euclidean distance and the family of Mahalanobis distances, in which the former is a trivial instance of the latter when Ω is set to the identity matrix, is analogous to that between the total variation distance and optimal transport distances, in which the former is also a trivial instance of the latter where all distances between features are uniformly set to 1. The two families of distances evolve in related albeit completely different sets of distances, just like the l_1 and l_2 norms describe different geometries. An illustration of this can be found in Figure 4 provided earlier in this paper, where the Euclidean and the total variation distances are compared with their parameterized counterparts. Both total variation and optimal transport distances have *piecewise linear* level sets, whereas the Euclidean and Mahalanobis distances have ellipsoidal level sets.

It is also worth mentioning that although Mahalanobis distances have been designed for general vectors in $\mathbb{R}^d$, and as a consequence can be applied to histograms, there is however, to our knowledge, no statistical theory which motivates their use on the probability simplex. This should be compared to the fact that there is a fairly large literature on optimal transport distances for probabilities, described by (Villani 2003, §7) and references therein.

5.3 Metric Learning in the Probability Simplex

Lebanon (2006) has proposed to learn a bin-to-bin distance in the probability simplex using a parametric family of distances parameterized by a histogram $\lambda \in \Sigma_{d-1}$ defined as

$$d_\lambda(r,c) = \arccos\left(\sum_{i=1}^{d}\sqrt{\frac{r_i\lambda_i}{r^T\lambda}}\sqrt{\frac{c_i\lambda_i}{c^T\lambda}}\right).$$

This formula can be simplified by using the perturbation operator proposed by Aitchison (1986, p.46):

$$\forall r,\lambda \in \Sigma_{d-1}, \quad r\odot\lambda \overset{\text{def}}{=} \frac{1}{r^T\lambda}(r_1\lambda_1,\cdots,r_d\lambda_d)^T.$$

Aitchison argues that the perturbation operation can be naturally interpreted as an addition operation in the simplex. Using this notation, the family of distances $d_\lambda(r,c)$ proposed by Lebanon can be seen as the standard Fisher metric applied to perturbed histograms $r\odot\lambda$ and $c\odot\lambda$,

$$d_\lambda(r,c) = \arccos\langle\sqrt{r\odot\lambda},\sqrt{c\odot\lambda}\,\rangle.$$

Using arguments related to the fact that a distance should vary according to the density of points described in a dataset, Lebanon (2006) proposes to learn this perturbation λ in an unsupervised context, by only considering histograms but no other side-information.

More recently, Kedem et al. (2012) have proposed non-linear metric learning techniques, and focus more specifically on parameterized χ_2 distances defined as $d^P_{\chi_2}(r,c) = d_{\chi_2}(Pr,Pc)$ where P can be any stochastic matrix P with unit row sums. We also note that, a few months after the publication on the arxiv of an early version of our paper, Wang and Guibas (2012) have proposed an algorithm that is very similar to ours, with the notable difference that they do not take into account metric constraints for the ground metric.

6. Experiments

We provide in this section a few details on the practical implementation of Algorithms 1, 2 and 3. We follow by presenting empirical evidence that ground metric learning improves upon other state-of-the-art metric learning techniques when considered on normalized histograms of low dimensions, albeit at a substantial computational cost.

6.1 Implementation Notes

Algorithm 1 builds upon the computation of several optimal transport problems. We use the *CPLEX Matlab API* implementation of network flows to that effect. Using directly the API is faster than calling the *CPLEX matlab toolbox* or the *Mosek* solver. These benefits come from the fact that only the constraint vector in Problem (4) needs to be updated at each iteration of the first loop of Algorithm 1. We use the *metricNearness* toolbox released online by Suvrit Sra to carry out both the projections of each inner loop iteration of Algorithm 2 and the last step of Algorithm 3.

6.2 Distances used in this benchmark

We consider five distances in this benchmark. Three classic bin-to-bin distances, Mahalanobis distances with different learning schemes and the optimal transport distance coupled with ground metric learning. *Bin-to-bin distances* We consider the l_1, l_2 and Hellinger distances on histograms,

$$l_1(r,c) = \|r-c\|_1, \quad l_2(r,c) = \|r-c\|_2, \quad \mathcal{H}(r,c) = \|\sqrt{r}-\sqrt{c}\|_2,$$

where $\sqrt{r}$ is the vector whose coordinates are the squared roots of each coordinate of r.

6.2.1 Mahalanobis distances

We use the publicly available implementations of LMNN (Weinberger and Saul 2009) and ITML (Davis et al. 2007) to learn Mahalanobis distances for each task. We run both algorithms with default settings, that is $k = 3$ for LMNN and $k = 4$ for ITML. We use these algorithms on the Hellinger representations $\{\sqrt{r_i},\ i = 1,\dots,n\}$ of all histograms originally in the training set using the element-wise square root. We have considered this representation because the Euclidean distance between the Hellinger representations of two histograms corresponds exactly to the Hellinger distance (Amari and Nagaoka 2001, p.57). Since the Mahalanobis distance builds upon the Euclidean distance, we argue that this representation is more adequate to learn Mahalanobis metrics in the probability simplex. This observation is confirmed in all of our experimental results, where Mahalanobis metric learning approaches perform consistently better with the Hellinger transformation (see for instance the results reported in Figure 7).

6.2.2 Optimal Transport Distances with Ground Metric Learning

We learn ground metrics using the following settings. The neighborhood parameter k is set to 3 to be directly comparable to the default parameter setting of ITML and LMNN. In each classification task, and for two images r_i and r_j, the corresponding weight ω_{ij} is set to

$1/nk$ if both histograms come from the same class and to $-1/nk$ if they come from different classes. The subgradient stepsize t_0 of Algorithm 2 is set to $= 0.1$, guided by preliminary experiments and by the fact that, because of the normalization of the weights ω_{ij}, both the current iteration M_k in Algorithm 2 and subgradients γ_+ or γ_- all have the same 1-norms.

We carry out a minimum of 24 subgradient steps in each inner loop and set $q_{\max}$ to 80. Each inner loop is terminated when the objective does not progress more than 0.75% every 8 steps, or when q reaches $q_{\max}$. We carry out a maximum of 20 outer loop iterations. With these settings, the algorithm takes about 300 steps to converge (Figures 8 and 9), which, using a single Xeon 2.6Ghz core, 60 training points and $d = 128$ (the experimental setting considered below) takes about 900 seconds. The main computational bottleneck of the algorithm comes from the repeated computation of optimal transports. LMNN and ITML parameterized with default settings converge much faster, in about 2 and 30 seconds respectively.

6.3 Binary Classification

We study in this section the performance of ground metric learning when coupled with a nearest neighbor classifier on binary classification tasks generated with the Caltech-256 database.

6.3.1 Experimental Setting

We sample randomly 80 images for each of the 256 images classes[2] of the Caltech-256 database. Each image is represented as a normalized histogram of GIST features (Oliva and Torralba 2001, Douze et al. 2009), obtained using an implementation provided by the INRIA-LEAR team[3]. These features describe 8 edge directions at mid-resolution computed for each patch of a 4×4 grid on each image. Each feature histogram is of dimension $d = 8 \times 4 \times 4 = 128$ and subsequently normalized to sum to one.

We select randomly 1,000 distinct pairs of classes among the 256 classes available in the dataset to form as many binary classification tasks. For each pair, we split the $80 + 80$ available points into $30+30$ points to train distance parameters and $50+50$ points to form a test set. This amounts to having $n = 60$ training points following the notations introduced in Section 3.1. We consider in the following κ nearest neighbors approaches. Note that the neighborhood size κ and the parameter k used in metric learning approaches need not be the same. In our experiments κ varies, whereas k is always kept fixed, as detailed in Section 6.2.

6.3.2 Results

The most important results of this experimental section are summarized in Figure 6, which displays, for all considered distances, their average recall accuracy on the test set and the average classification error using a κ-nearest neighbor classifier. These quantities are averaged over 1,000 binary classifications. In this figure, GML paired with the the optimal transport distance d_M is shown to provide, on average, the best performance with three

[2] we do not consider the *clutter* class

[3] `http://lear.inrialpes.fr/software`

different metrics: the leftmost plot considers retrieval performance for test points and shows that, for each point considered on its own, GML-EMD selects on average more training points from the same class as closest neighbors than any other distance. The performance gap between GML-EMD and competing distances increases significantly as the number of retrieved neighbors is itself increased. The middle plot displays the average error over all 1,000 tasks of a κ-nearest neighbor classification algorithm when considered with all distances for varying values of κ. The rightmost plot studies these errors in more detail for the case where the neighborhood parameter κ of nearest neighbors is 3. In this case too, GML combined with EMD fares significantly better than competing distances.

Figure 8 illustrates the empirical behavior of our descent algorithm. This plot displays 40 sample objective curves among the 1,000 computed to obtain the results above. The bumps that appear regularly on these curves correspond to the first update carried out after the linearization of the concave part of the objective. These results were obtained by setting the initial matrix to $M_{\mathbb{1}}$.

It is also worth mentioning as a side remark that the l_2 distance does not perform as well as the l_1 or Hellinger distances on these datasets, which validates our earlier statement that the Euclidean geometry is usually a poor choice to compare histograms directly. This intuition is further validated in Figure 7, where Mahalanobis learning algorithms are shown to perform significantly better when they use the Hellinger representation of histograms.

Finally, Figure 9 describes the evolution of the average *test* error for two initial ground metrics, $M_{\mathbb{1}}$ and that which builds upon independence tables (Algorithm 3). Two conclusions can be drawn from this plot: First, independence tables provide on average a better initialization of the algorithm if only the first iterations of the algorithm are taken into account. However, this advantage seems to vanish as the number of subgradient descent iterations increases. Second, our algorithm does not seem to suffer from overfitting on average, since the average error rate is a decreasing curve of the total number of iterations and does not seem to increase up to termination.

6.4 Multiclass Classification

We follow our experimental evaluation of ground metric learning by considering this time 6 multiclass classification datasets that consider text and image data.

6.4.1 Experimental Setting

The properties of the datasets and parameters used in our experiments are summarized in Table 1. The dimensions of the features have been kept low to ensure that the computation of optimal transports are tractable. We follow the recommended train/test splits for these datasets. If they are not provided, we split the datasets arbitrarily to form features using either LDA (Blei et al. 2003) or SIFT features (Lowe 1999). We then generate 5 random splits with the same balance to compute average accuracies over the entire dataset.

6.4.2 Results

Figure 10 details the results for these 6 experiments, and show that GML coupled with EMD is at least equivalent or improves on the best techniques considered in our benchmark. These results also illustrate that the performance of Mahalanobis learning (LMNN in this case) is

Table 1: Multiclass classification datasets and their parameters.

Dataset	#Train	#Test	#Class	Feature	#Dim
20 News Group	600	19397	20	Topic Model (LDA)	100
Reuters	500	9926	10	Topic Model(LDA)	100
MIT Scene	800	800	8	SIFT	100
UIUC Scene	1500	1500	15	SIFT	100
OXFORD Flower	680	680	17	SIFT	100
CALTECH-101	3060	2995	102	SIFT	100

greatly improved by considering the Hellinger representation of histograms, and not their original representation as vectors of the simplex.

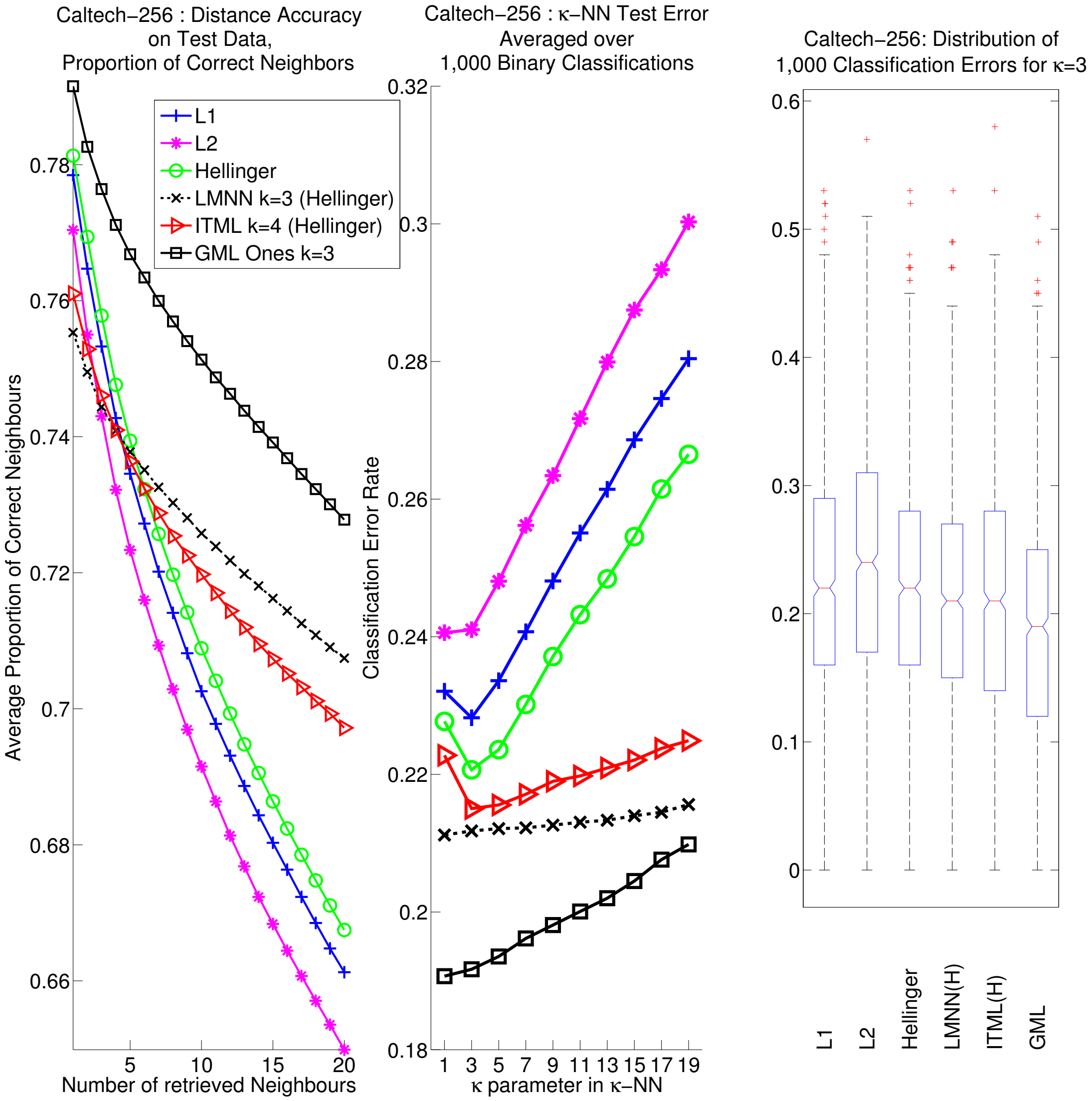

Figure 6: (left) Accuracy of each considered distance on the test set as measured by the average proportion, for each datapoint in the test set, of points coming from the same class within its κ nearest neighbors. These proportions were averaged over 1,000 binary classification problems randomly chosen among the $\binom{256}{2}$ possible. We use 40 test points from each class for each experiment, namely 80 test points. The ground metric in GML and Mahalanobis matrices in ITML and LMNN have been learned using a train set of $30 + 30$ points. (middle) κ-NN classification error using the same distances. These results show average κ-NN error over 1,000 classification tasks depending on the value of κ. A more detailed picture for the case $\kappa = 3$ is provided with boxplots of all $1,000$ errors (right).

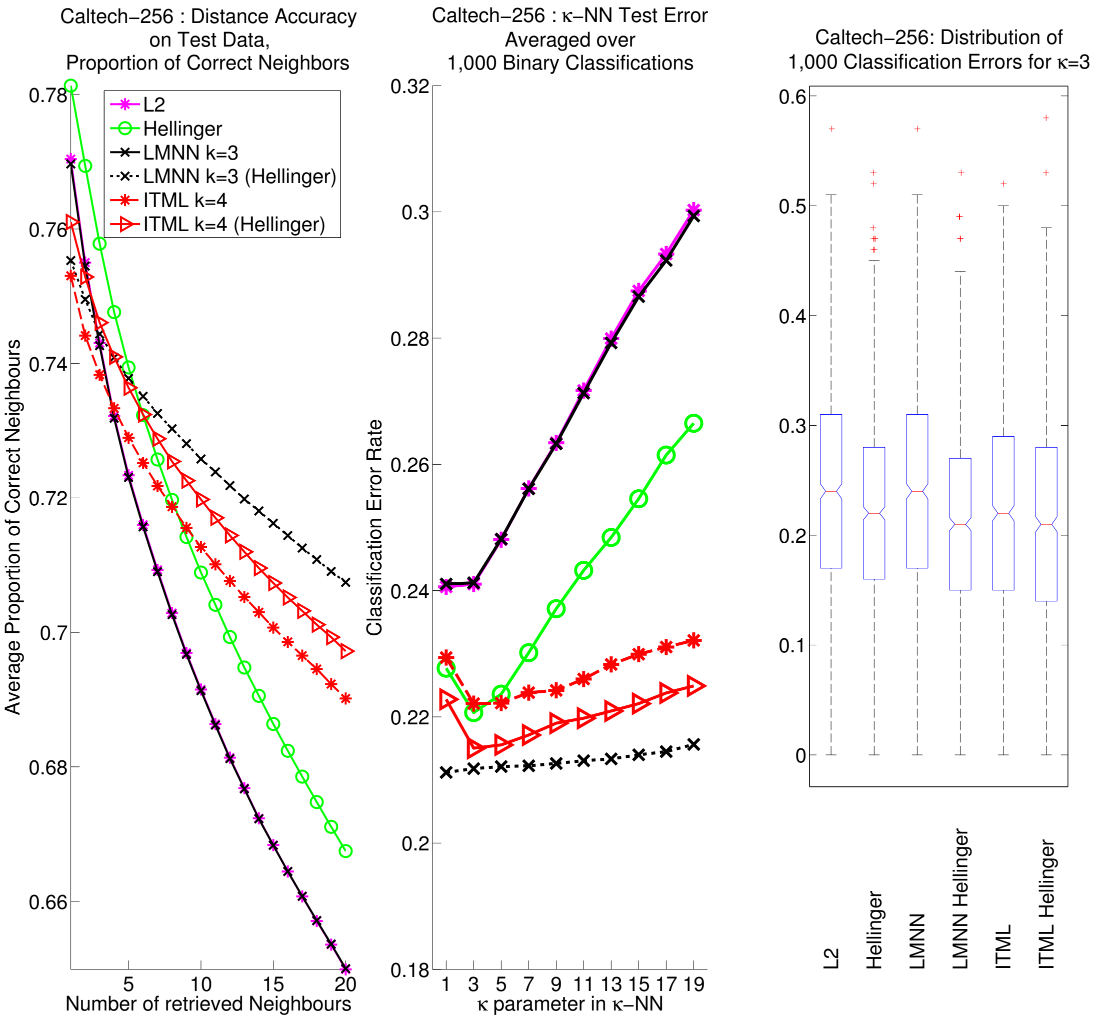

Figure 7: The experimental setting in this figure is identical to that of Figure 6, except that only two different versions of LMNN and ITML are compared with the Hellinger and Euclidean distances. This figure supports our claim in Section 6.2.1 that Mahalanobis learning methods work better using the Hellinger representation of histograms, $\{\sqrt{r_i},\ i = 1, \ldots, n\}$, rather than their straightforward representation in the simplex $\{r_i\}_{i=1,\ldots,n}$.

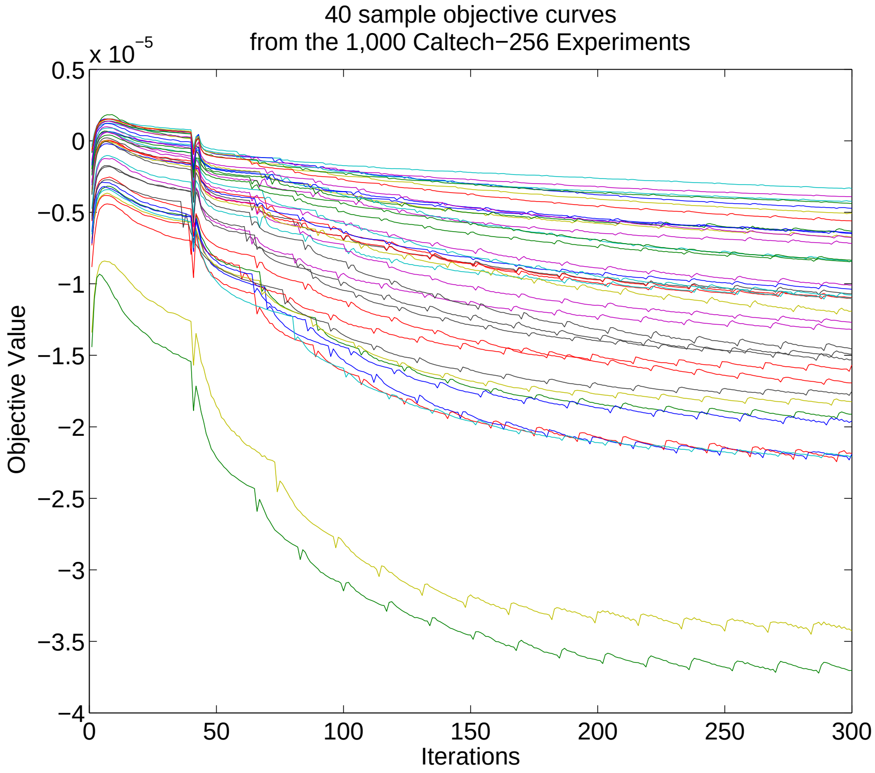

Figure 8: 40 sample objective curves randomly selected among the 1,000 binary classification tasks run on the Caltech-256 dataset. The initial point used here is the matrix $M_{\mathbb{1}}$ of ones and zero diagonal. The very first bumps usually observed in the first iterations agree with our empirical findings on empirical test error displayed in Figure 9 which illustrate that the very first radients that are applied are usually not informative and result momentarily in an objective increase.

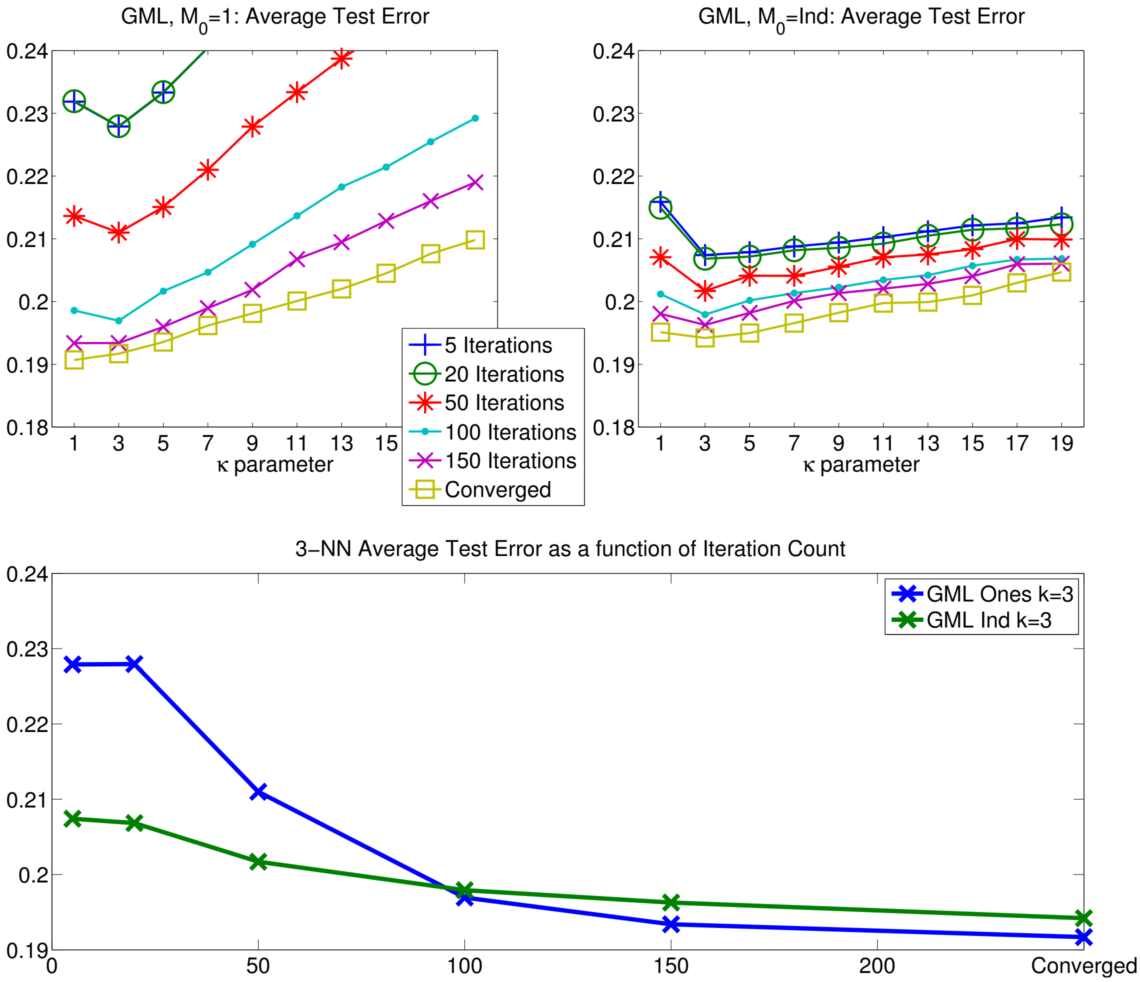

Figure 9: Average κ-nearest neighbor *test* error for GML using either the matrix of ones (top left) or the independent table (top right) described in Section 4.3. As can be seen for $\kappa = 3$ (bottom), initializing the algorithm with $M_{\mathbb{1}}$ performs worse than independence tables for a low iteration count. Yet this competitive advantage is reversed above a few iterations, as the algorithm converges. This figure also seems to indicate that, on average, the algorithm does not overfit the data since the average test error seems to decrease monotonically with the number of iterations, and becomes flat after 200 iterations. The experimental setting is identical to that of Figure 6.

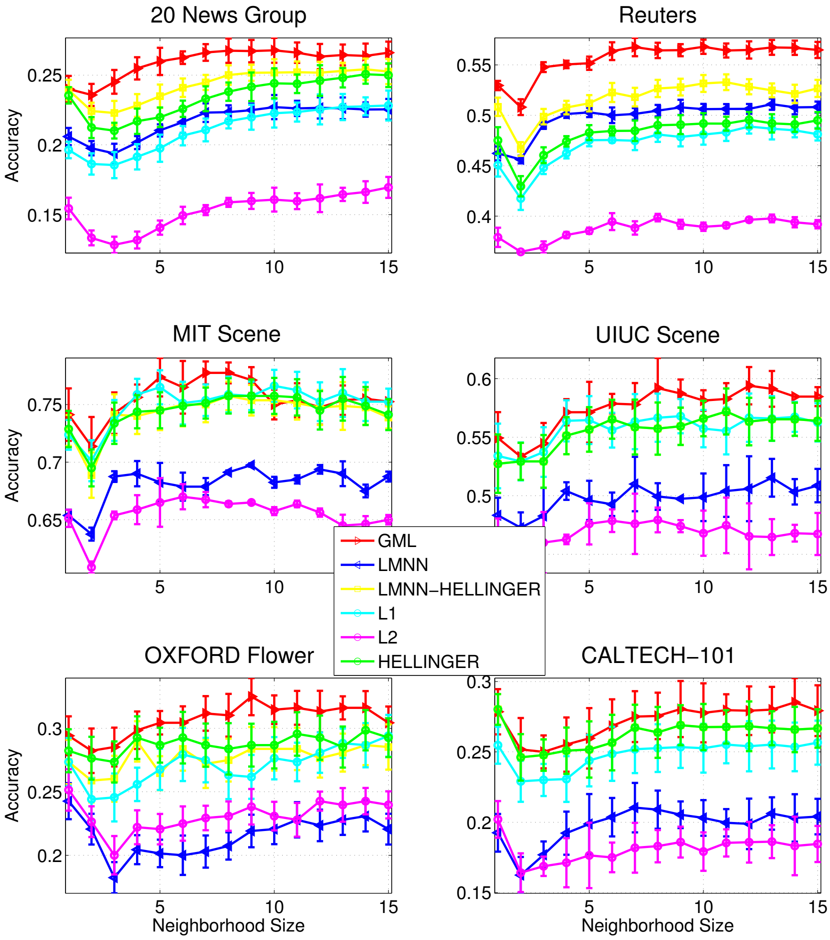

Figure 10: κ-nearest neighbor performance for different distances on multi-class problems. Performance is averaged over 5 repeats, whose variability is illustrated with error bars. Errors are reported over varying κ nearest neighbor parameters. Our benchmark considers three classical distances, l_1, l_2 and Hellinger, and their respective learned counterparts: GML paired with the transport distance initialized with the matrix $M_{\mathbb{1}}$, classic LMNN and LMNN on the Hellinger representation.

7. Conclusion and Future Work

We have proposed in this paper an approach to tune adaptively the unique parameter of optimal transport distances, the ground metric, given a training dataset of histograms. This approach can be applied on any type of features, as long as a set of histograms along with side-information, typically labels, are provided for the algorithm to learn a good candidate for the ground metric. The algorithms proceeds with a projected subgradient descent to minimize approximately a criterion that is a difference of polyhedral convex functions. We propose two initial points to initialize this algorithm, and show that our approach provides, when compared to other competing distances, a superior average performance for a large set of image binary classification tasks using GIST features histograms, as well as different multiclass classification tasks. This improvement comes, however, with a heavy computational price tag.

Our benchmark experiments only contain low-dimensional descriptors. We chose such small dimensions because it is well known that optimal transport distances do not scale well for higher dimensions. That being said, the problem of speeding up the computation of optimal transport distances by considering restrictions on ground metrics has attracted significant attention. Ling and Okada (2007), Gudmundsson et al. (2007), Pele and Werman (2009), Ba et al. (2011) have all recently argued that this computation can be dramatically sped up when the ground metric matrix has a certain structure. For instance, Pele and Werman (2009) have shown that the computational speed of earth mover's distances can be significantly accelerated when the ground metric is thresholded above a certain level. Ground metrics that follow such constraints are attractive because they result in transport problems which are provably faster to compute. Our work in this paper suggests on the other hand that the content (and not the structure) of the ground metric can be learned to improve classification accuracy. We believe that the combination of these two viewpoints could result in optimal transport distances that are both adapted to the task at hand and fast to compute. A strategy to achieve both goals would be to enforce such structural constraints on candidate metrics M when looking for minimizers of criteria C_k. We also believe that the recent proposal of Sinkhorn distances (Cuturi 2013) may provide the necessary speed-ups to make our approach more scaleable regardless of the structure of the ground metric.

Acknowledgements

The authors would like to thank anonymous reviewers and the editor for stimulating comments. MC would like to thank Zaid Harchaoui and Alexandre d'Aspremont for fruitful discussions, as well as Tam Le for his help in preparing some of the experiments.

Appendix

Proof (Theorem 1) Symmetry and definiteness of the distance are easy to prove: since M has a null diagonal, $d_M(x,x)=0$, with corresponding optimal transport matrix $X^\star = \operatorname{diag}(x)$; by the positivity of all off-diagonal elements of M, $d_M(x,y)>0$ whenever $x \neq y$; by symmetry of M, d_M is itself a symmetric function in its two arguments. To prove the triangle inequality, Villani (2003, Theorem 7.3) uses the gluing lemma. We provide here a self-contained version of this proof which provides an explicit formulation for the gluing lemma in the discrete case. Let $x, y, z \in \Sigma_d$. Let P and Q be two optimal solutions of the transport problems between x and y, and y and z respectively. Let S be the $d \times d \times d$ tensor whose coefficients are defined as

$$s_{ijk} \overset{\text{def}}{=} \frac{p_{ij} q_{jk}}{y_j},$$

for all indices j such that $y_j > 0$. For indices j such that $y_j = 0$, the corresponding values s_{ijk} are set to 0. S is a probability measure on $\{1, \ldots, d\}^3$, as a direct consequence of the fact that the $d \times d$ matrix $S_{i \cdot k} \overset{\text{def}}{=} [\sum_j s_{ijk}]_{ik}$ is a transport matrix between x and z and thus sums to 1. Indeed,

$$\sum_i \sum_j s_{ijk} = \sum_j \sum_i \frac{p_{ij} q_{jk}}{y_j} = \sum_j \frac{q_{jk}}{y_j} \sum_i p_{ij} = \sum_j \frac{q_{jk}}{y_j} y_j = \sum_j q_{jk} = z_k \text{ (column sums)}$$
$$\sum_k \sum_j s_{ijk} = \sum_j \sum_k \frac{p_{ij} q_{jk}}{y_j} = \sum_j \frac{p_{ij}}{y_j} \sum_k q_{jk} = \sum_j \frac{p_{ij}}{y_j} y_j = \sum_j p_{ij} = x_i \text{ (row sums)}$$

To obtain the triangle inequality, notice that $S_{i \cdot k}$ being a matrix of $U(x, z)$ we can write:

$$\begin{aligned}
d_M(x, z) &= \min_{X \in U(x,z)} \langle X, M \rangle \\
&\leq \langle S_{i \cdot k}, M \rangle = \sum_{ik} m_{ik} \sum_j \frac{p_{ij} q_{jk}}{y_j} \leq \sum_{ijk} (m_{ij} + m_{jk}) \frac{p_{ij} q_{jk}}{y_j} \\
&= \sum_{ijk} m_{ij} \frac{p_{ij} q_{jk}}{y_j} + m_{jk} \frac{p_{ij} q_{jk}}{y_j} \\
&= \sum_{ij} m_{ij} p_{ij} \sum_k \frac{q_{jk}}{y_j} + \sum_{jk} m_{jk} q_{jk} \sum_i \frac{p_{ij}}{y_j} \\
&= \sum_{ij} m_{ij} p_{ij} + \sum_{jk} m_{jk} q_{jk} = d_M(x, y) + d_M(y, z),
\end{aligned}$$

where we have used the triangle inequality for M at the end of the second line. ■

References

R. Ahuja, T. Magnanti, and J. Orlin. *Network Flows: Theory, Algorithms and Applications.* Prentice Hall, 1993.

J. Aitchison. *The Statistical Analysis of Compositional Data.* Chapman & Hall, 1986.

J. Aitchison and J. Egozcue. Compositional data analysis: Where are we and where should we be heading? *Mathematical Geology*, 37(7):829–850, 2005.

S.-I. Amari and H. Nagaoka. *Methods of Information Geometry.* AMS vol. 191, 2001.

K. Ba, H. Nguyen, H. Nguyen, and R. Rubinfeld. Sublinear time algorithms for earth movers distance. *Theory of Computing Systems*, 48(2):428–442, 2011.

A. Barvinok. What does a random contingency table look like? *Combinatorics, Probability and Computing*, 19(04):517–539, 2010.

A. Bellet, A. Habrard, and M. Sebban. A survey on metric learning for feature vectors and structured data. *arXiv:1306.6709*, 2013.

D. Bertsimas and J. Tsitsiklis. *Introduction to Linear Optimization.* Athena Scientific, 1997.

D. Blei and J. Lafferty. Topic models. *Text Mining: Classification, Clustering, and Applications*, 10:71, 2009.

D. Blei, A. Ng, and M. Jordan. Latent Dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022, 2003.

S. Boyd and L. Vandenberghe. *Convex Optimization.* Cambridge University Press, 2004.

J. Brickell, I. Dhillon, S. Sra, and J. Tropp. The metric nearness problem. *SIAM Journal of Matrix Analysis and Applications*, 30(1):375–396, 2008.

R. A. Brualdi. *Combinatorial Matrix Classes*, volume 108. Cambridge University Press, 2006.

O. Chapelle, P. Haffner, and V. Vapnik. SVMs for histogram based image classification. *IEEE Transactions on Neural Networks*, 10(5):1055, Sept. 1999.

M. Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems 26*, pages 2292–2300. 2013.

J. Davis, B. Kulis, P. Jain, S. Sra, and I. Dhillon. Information-theoretic metric learning. In *Proceedings of the 24th International Conference on Machine Learning*, pages 209–216. ACM, 2007.

M. Deza and E. Deza. *Encyclopedia of Distances.* Springer Verlag, 2009.

P. Diaconis and B. Efron. Testing for independence in a two-way table: new interpretations of the chi-square statistic. *The Annals of Statistics*, 13(3):845–913, 1985.

M. Douze, H. Jégou, H. Sandhawalia, L. Amsaleg, and C. Schmid. Evaluation of GIST descriptors for web-scale image search. In *Proceedings of the ACM International Conference on Image and Video Retrieval.* Article 19, ACM, 2009.

L. Ford and Fulkerson. *Flows in Networks.* Princeton University Press, 1962.

A. Frangioni, A. Lodi, and G. Rinaldi. New approaches for optimizing over the semimetric polytope. *Mathematical programming*, 104(2):375–388, 2005.

I. Good. Maximum entropy for hypothesis formulation, especially for multidimensional contingency tables. *The Annals of Mathematical Statistics*, pages 911–934, 1963.

J. Gudmundsson, O. Klein, C. Knauer, and M. Smid. Small manhattan networks and algorithmic applications for the earth movers distance. In *Proceedings of the 23rd European Workshop on Computational Geometry*, pages 174–177, 2007.

T. Joachims. *Learning to Classify Text Using Support Vector Machines: Methods, Theory, and Algorithms.* Kluwer Academic Publishers, 2002.

L. Kantorovich and G. Rubinshtein. On a space of totally additive functions. *Vestn Lening. Univ.*, 13:52–59, 1958.

H. Kashima, K. Tsuda, and A. Inokuchi. Marginalized kernels between labeled graphs. In *Proceedings of the 20th International Conference on Machine Learning*, pages 321–328, 2003.

D. Kedem, S. Tyree, K. Weinberger, F. Sha, and G. Lanckriet. Non-linear metric learning. In *Advances in Neural Information Processing Systems 25*, pages 2582–2590, 2012.

B. Kulis. Metric learning: A survey. *Foundations & Trends in Machine Learning*, 5(4): 287–364, 2012.

S. Lauritzen. *Lectures on Contingency Tables.* Aalborg Univ. Press, 1982.

G. Lebanon. Metric learning for text documents. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):497–508, 2006.

C. Leslie, E. Eskin, and W. S. Noble. The spectrum kernel: a string kernel for svm protein classific ation. In *Proceedings of the Pacific Symposium on Biology 2002*, pages 564–575, 2002.

E. Levina and P. Bickel. The earth mover's distance is the Mallows distance: some insights from statistics. In *Proceedings of the Eighth IEEE International Conference on Computer Vision*, volume 2, pages 251–256. IEEE, 2001.

H. Ling and K. Okada. An efficient earth mover's distance algorithm for robust histogram comparison. *IEEE transactions on Pattern Analysis and Machine Intelligence*, pages 840–853, 2007.

D. Lowe. Object recognition from local scale-invariant features. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 1150 –1157 vol.2, 1999.

C. Mallows. A note on asymptotic joint normality. *The Annals of Mathematical Statistics*, pages 508–515, 1972.

A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001.

O. Pele and M. Werman. Fast and robust earth mover's distances. In *Proceedings of the International Conference on Computer Vision'09*, 2009.

S. Rachev. *Probability Metrics and the Stability of Stochastic Models.* Wiley series in probability and mathematical statistics: Applied probability and statistics. Wiley, 1991.

S. Rachev and L. Rüschendorf. *Mass Transportation Problems: Theory*, volume 1. Springer Verlag, 1998.

T. Rockafellar. *Convex Analysis.* Princeton University Press, 1970.

Y. Rubner, L. Guibas, and C. Tomasi. The earth movers distance, multi-dimensional scaling, and color-based image retrieval. In *Proceedings of the ARPA Image Understanding Workshop*, pages 661–668, 1997.

Y. Rubner, C. Tomasi, and L. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40, 2000.

M. Swain and D. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1): 11–32, 1991.

A. Vershik. Kantorovich metric: initial history and little-known applications. *Journal of Mathematical Sciences*, 133(4):1410–1417, 2006.

C. Villani. *Topics in Optimal Transportation*, volume 58. AMS Graduate Studies in Mathematics, 2003.

F. Wang and L. J. Guibas. Supervised earth movers distance learning and its computer vision applications. In *Computer Vision–ECCV 2012*, pages 442–455. Springer, 2012.

K. Weinberger and L. Saul. Distance metric learning for large margin nearest neighbor classification. *The Journal of Machine Learning Research*, 10:207–244, 2009.

K. Weinberger, J. Blitzer, and L. Saul. Distance metric learning for large margin nearest neighbor classification. In *Advances in Neural Information Processing Systems 18*, pages 1473–1480, 2006.

E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell. Distance metric learning with application to clustering with side-information. In *Advances in Neural Information Processing Systems 15*, pages 505–512. MIT Press, 2003.