Geometric Dataset Distances via Optimal Transport

David Alvarez-Melis
Microsoft Research, New England
alvarez.melis@microsoft.com

Nicolò Fusi
Microsoft Research, New England
nfusi@microsoft.com

Abstract

The notion of task similarity is at the core of various machine learning paradigms, such as domain adaptation and meta-learning. Current methods to quantify it are often heuristic, make strong assumptions on the label sets across the tasks, and many are architecture-dependent, relying on task-specific optimal parameters (*e. g.,* require training a model on each dataset). In this work we propose an alternative notion of distance between datasets that (i) is model-agnostic, (ii) does not involve training, (iii) can compare datasets even if their label sets are completely disjoint and (iv) has solid theoretical footing. This distance relies on optimal transport, which provides it with rich geometry awareness, interpretable correspondences and well-understood properties. Our results show that this novel distance provides meaningful comparison of datasets, and correlates well with transfer learning hardness across various experimental settings and datasets.

1 Introduction

A key hallmark of machine learning practice is that labeled data from the application of interest is usually scarce. For this reason, there is vast interest in methods that can combine, adapt and transfer knowledge across datasets and domains. Entire research areas are devoted to these goals, such as domain adaptation, transfer-learning and meta-learning. A fundamental concept underlying all these paradigms is the notion of *distance* (or more generally, *similarity*) between datasets. For instance, transferring knowledge across similar domains should intuitively be easier than across distant ones. Likewise, given a choice of various datasets to pretrain a model on, it would seem natural to choose the one that is closest to the task of interest.

Despite its evident usefulness and apparent simpleness, the notion of distance between datasets is an elusive one, and quantifying it efficiently and in a principled manner remains largely an open problem. Doing so requires solving various challenges that commonly arise precisely in the settings for which this notion would be most useful, such as the ones mentioned above. For example, in supervised machine learning settings the datasets consist of both features and labels, and while defining a distance between the former is often —though not always— trivial, doing so for the labels is far from it, particularly if the label-sets across the two tasks are not identical (as is often the case for off-the-shelf pretrained models).

Current approaches to transfer learning that seek to quantify dataset similarity circumvent these challenges in various ingenious, albeit often heuristic, ways. A common approach is to compare the dataset via proxies, such as the learning curves of a pre-specified model [37] or its optimal parameters [2, 32] on a given task, or by making strong assumptions on the similarity or co-occurrence of labels across the two datasets [50]. Most of these approaches lack guarantees, are highly dependent on the probe model used, and require training a model to completion (*e. g.,* to find optimal parameters) on each dataset being compared. On the opposite side of the spectrum are principled notions of discrepancy between domains [9, 41], which nevertheless are often not computable in practice, or do not scale to the type of datasets used in machine learning practice.

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

In this work, we seek to address some of these limitations by proposing an alternative notion of distance between datasets. At the heart of this approach is the use of optimal transport (OT) distances [52] to compare distributions over feature-label pairs in a geometrically-meaningful and principled way. In particular, we propose a hybrid Euclidean-Wasserstein distance between feature-label pairs across domains, where labels themselves are modeled as distributions over features vectors. As a consequence of this technique, our framework allows for comparison of datasets *even if their label sets are completely unrelated or disjoint*, as long as a distance between their features can be defined. This notion of distance between labels, a by-product of our approach, has itself various potential uses, *e. g.,* to optimally sub-sample classes from large datasets for more efficient pretraining.

In summary, we make the following contributions:

- We introduce a principled, flexible and efficiently computable notion of distance between datasets
- We propose algorithmic strategies to scale up computation of this distance to very large datasets
- We provide extensive empirical evidence that this distance is highly predictive of transfer learning success across various domains, tasks and data modalities

2 Related Work

Discrepancy Distance Various notions of (dis)similarity between data distributions have been proposed in the context of domain adaptation, such as the d_A [9] and discrepancy distances[1] [41]. These discrepancies depend on a loss function and hypothesis (*i. e.,* predictor) class, and quantify dissimilarity through a supremum over this function class. The latter discrepancy in particular has proven remarkably useful for proving generalization bounds for adaptation [13], and while it can be estimated from samples, bounding the approximation quality relies on quantities like the VC-dimension of the hypothesis class, which might not be always known or easy to compute.

Dataset Distance via Parameter Sensitivity The Fisher information metric is a classic notion from information geometry [6, 8] that characterizes a parametrized probability distribution locally through the sensitivity of its density to changes in the parameters. In machine learning, it has been used to analyze and improve optimization approaches [7] and to measure the capacity of neural networks [39]. In recent work, Achille et al. [2] use this notion to construct vector representations of tasks, which they then use to define a notion of similarity between these. They show that this notion recovers taxonomic similarities and is useful in meta-learning to predict whether a certain feature extractor will perform well in a new task. While this notion shares with ours its agnosticism of the number of classes and their semantics, it differs in the fact that it relies on a probe network trained on a specific dataset, so its geometry is heavily influenced by the characteristics of this network. Besides the Fisher information, a related information-theoretic notion of complexity that can be used to characterize tasks is the Kolmogorov Structure Function [38], which Achille et al. [1] use to define a notion of *reachability* between tasks.

Optimal Transport-based distributional distances The general idea of representing complex objects via distributions, which are then compared through optimal transport distances, is an active area of research. Also driven by the appeal of their closed-form Wasserstein distance, Muzellec and Cuturi [44] propose to embed objects as elliptical distributions, which requires differentiating through these distances, and discuss various approximations to scale up these computations. Frogner et al. [25] extend this idea but represent the embeddings as discrete measures (*i. e.,* point clouds) rather than Gaussian/Elliptical distributions. Both of these works focus on embedding and consider only within-dataset comparisons. Also within this line of work, Delon and Desolneux [19] introduce a Wasserstein-type distance between Gaussian mixture models. Their approach restricts the admissible transportation couplings themselves to be Gaussian mixture models, and does not directly model label-to-label similarity. More generally, the Gromov-Wasserstein distance [43] has been proposed to compare collections across different domains [4, 42], albeit leveraging only features, not labels.

Hierarchical OT distances The distance we propose can be understood as a hierarchical OT distance, *i. e.,* one where the ground metric itself is defined through an OT problem. This principle has been explored in other contexts before. For example, Yurochkin et al. [55] use a hierarchical OT distance for document similarity, defining a inner-level distance between topics and a outer-level distance between documents using OT. Dukler et al. [22] on the other hand use a nested Wasserstein distance as a loss for generative model training, motivated by the observation that the Wasserstein

[1] Despite its name, this discrepancy is not a distance in general.

distance is better suited to comparing images than the usual pixel-wise L_2 metric used as ground metric. Both the goal and the actual metric used by these approaches differs from ours.

Optimal Transport with Labels Using label information to guide the optimal transport problem towards class-coherent matches has been explored before, *e. g.,* by enforcing group-norm penalties [15] or through submodular cost functions [5]. These works are focused on the unsupervised domain adaptation setting, so their proposed modifications to the OT objective use only label information from one of the two domains, and even then, do so without explicitly defining a metric between these. Furthermore, they do not lead to proper distances, and these works deal with a single static pair of tasks, so they lack analysis of the distance across multiple source and target datasets. Closest to this work is JDOT [14] and its extension by Damodaran et al. [17], which use a hybrid feature-label transportation cost that quantifies discrepancy of labels through a classification loss (e.g., hinge-loss). As a consequence, this approach requires the two distributions share the exact same label set, might not yield a true metric depending on the loss chosen, and requires careful scaling of the two components of the cost.

3 Background on Optimal Transport

Optimal transport (OT) is a powerful and principled approach to compare probability distributions with deep theoretical foundations [52, 53] and desirable computational properties [46]. It considers a complete and separable metric space $\mathcal{X}$, along with probability measures $\alpha \in \mathcal{P}(\mathcal{X})$ and $\beta \in \mathcal{P}(\mathcal{X})$. These can be continuous or discrete measures, the latter often used in practice as empirical approximations of the former whenever working in the finite-sample regime. The Kantorovich formulation [31] of the transportation problem reads:

$$\mathrm{OT}(\alpha, \beta) \triangleq \min_{\pi \in \Pi(\alpha,\beta)} \int_{\mathcal{X}\times\mathcal{X}} c(x, y)\, \mathrm{d}\pi(x, y), \tag{1}$$

where $c(\cdot, \cdot) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$ is a cost function (the "ground" cost), and the set of couplings $\Pi(\alpha, \beta)$ consists of joint distributions over the product space $\mathcal{X} \times \mathcal{X}$ with marginals α and β:

$$\Pi(\alpha, \beta) \triangleq \{\pi \in \mathcal{P}(\mathcal{X}\times\mathcal{X}) \mid P_{1\#}\pi = \alpha, P_{2\#}\pi = \beta\}. \tag{2}$$

Whenever $\mathcal{X}$ is equipped with a metric $d_\mathcal{X}$, it is natural to use it as ground cost, *e. g.,* $c(x, y) = d_\mathcal{X}(x, y)^p$ for some $p \geq 1$. In such case, $\mathrm{W}_p(\alpha, \beta) \triangleq \mathrm{OT}(\alpha, \beta)^{1/p}$ is called the p-Wasserstein distance. The case $p = 1$ is also known as the Earth Mover's Distance [48].

The measures α and β are rarely known in practice. Instead, one has access to finite samples $\{\mathbf{x}^{(i)}\} \in \mathcal{X}, \{\mathbf{y}^{(j)}\} \in \mathcal{X}$, which implicitly define discrete measures $\alpha = \sum_{i=1}^n \mathbf{a}_i \delta_{\mathbf{x}^{(i)}}$ and $\beta = \sum_{i=1}^m \mathbf{b}_i \delta_{\mathbf{y}^{(j)}}$, where $\mathbf{a}$, $\mathbf{b}$ are vectors in the probability simplex, and the pairwise costs can be compactly represented as an $n \times m$ matrix $\mathbf{C}$, *i. e.,* $\mathbf{C}_{ij} = c(\mathbf{x}^{(i)}, \mathbf{y}^{(j)})$. In this case, Eq. (1) becomes a linear program, whose cubic complexity is often prohibitive. The entropy-regularized problem

$$\mathrm{OT}_\epsilon(\alpha, \beta) \triangleq \min_{\pi \in \Pi(\alpha,\beta)} \int_{\mathcal{X}\times\mathcal{X}} c(x, y)\, \mathrm{d}\pi(x, y) + \varepsilon \mathrm{H}(\pi \,|\, \alpha \otimes \beta), \tag{3}$$

where $\mathrm{H}(\pi \mid \alpha \otimes \beta) = \int \log(\mathrm{d}\pi / \,\mathrm{d}\alpha\, \mathrm{d}\beta)\, \mathrm{d}\pi$ is the relative entropy, can be solved much more efficiently —and with better sample complexity [28]— by using the Sinkhorn algorithm [3, 16], which enables a time/accuracy trade-off through ε. The *Sinkhorn divergence* [27], defined as

$$\mathrm{SD}_\varepsilon(\alpha, \beta) = \mathrm{OT}_\varepsilon(\alpha, \beta) - \frac{1}{2}\mathrm{OT}_\varepsilon(\alpha, \alpha) - \frac{1}{2}\mathrm{OT}_\varepsilon(\beta, \beta), \tag{4}$$

has many useful properties: it is positive, convex, and metrizes convergence of measures [24].

4 Optimal Transport between Datasets

The definition of *dataset* is notoriously inconsistent across the machine learning literature, sometimes referring only to features, or both features and labels. Here we are interested in supervised learning, so we define a dataset $\mathcal{D}$ as a set of feature-label pairs $(x, y) \in \mathcal{X} \times \mathcal{Y}$ over a certain feature space $\mathcal{X}$ and label set $\mathcal{Y}$. We will use the shorthand notations $z \triangleq (x, y)$ and $\mathcal{Z} \triangleq \mathcal{X} \times \mathcal{Y}$. Henceforth, we focus on the case of classification, so $\mathcal{Y}$ shall be a finite set. We consider two datasets $\mathcal{D}_A$ and $\mathcal{D}_B$, and assume, for simplicity, that their feature spaces have the same dimensionality, but

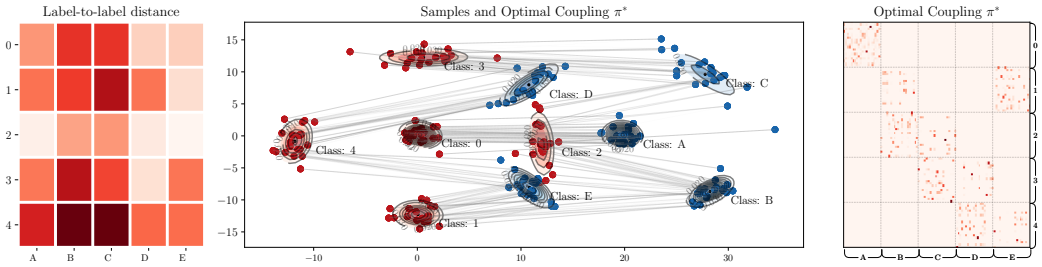


Figure 1: Our approach represents labels as distributions over features and computes Wasserstein distances between them (left). Combined with the usual metric between features, this yields a transportation cost between datasets. The optimal transport problem then characterizes the distance between them as the lowest possible cost to couple them (optimal coupling π^* shown on the right).

will discuss how to relax this assumption later on. On the other hand, we make no assumptions on the label sets $\mathcal{Y}_A$ and $\mathcal{Y}_B$ whatsoever. In particular, the classes these encode could be partially overlapping or related (*e. g.,* ImageNet and CIFAR-10) or completely disjoint (*e. g.,* CIFAR-10 and MNIST). Although not a formal assumption of our approach, it will be useful to think of the samples in these two datasets as being drawn from joint distributions $P_A(x, y)$ and $P_B(x, y)$, *i. e.,* $\mathcal{D}_A = \{(x_A^{(i)}, y_A^{(i)})\}_{i=1}^n \sim P_A(x, y)$ and $\mathcal{D}_B = \{(x_B^{(j)}, y_B^{(j)})\}_{j=1}^m \sim P_B(x, y)$.

Our goal is to define a distance $d(\mathcal{D}_A, \mathcal{D}_B)$ without relying on external models or parameters. The interpretation above, viewed in light of Section 3, suggests comparing these datasets by computing an OT distance between their joint distributions. However, casting problem (1) in this context requires a —crucial— component: a metric on $\mathcal{Z}$, *i. e.,* between pairs $(x, y), (x', y')$. If we had metrics on $\mathcal{X}$ and $\mathcal{Y}$, we could define a metric on $\mathcal{Z}$ as $d_{\mathcal{Z}}(z, z') = \big(d_{\mathcal{X}}(x, x')^p + d_{\mathcal{Y}}(y, y')^p\big)^{1/p}$, for $p \geq 1$. In most applications, $d_{\mathcal{X}}$ is readily available, *e. g.,* as the euclidean distance in the feature space. On the other hand, $d_{\mathcal{Y}}$ will rarely be so, particularly between labels from unrelated label sets (*e. g.,* between `cars` in one image domain and and `dogs` in the other). If we had some prior knowledge of the label spaces, we could use it to define a notion of distance between pairs of labels. However, in the challenging —but common— case where no such knowledge is available, the only information we have about the labels is their occurrence in relation to the feature vectors x. Thus, we can take advantage of the fact that we have a meaningful metric in $\mathcal{X}$ and use it to compare labels.

Formally, let $\mathsf{N}_{\mathcal{D}}(y) := \{x \in \mathcal{X} \mid (x, y) \in \mathcal{D}\}$ be the set of feature vectors with label y, and let n_y be its cardinality. With this, a distance between two labels y and y' could be defined as the distance between the centroids of $\mathsf{N}_{\mathcal{D}}(y)$ and $\mathsf{N}_{\mathcal{D}}(y')$. But representing the collections $\mathsf{N}_{\mathcal{D}}(y)$ only through their mean is likely too simplistic for real datasets. Ideally, we would represent labels through the *actual distribution* over the feature space that they define, namely, by the map $y \mapsto \alpha_y(X) \triangleq P(X \mid Y = y)$, of which $\mathsf{N}_{\mathcal{D}}(y)$ can be understood as a finite sample. If we use this representation, defining a distance between labels boils down to choosing a divergence between their associated distributions. Here again we argue that OT is an ideal choice, since it: (i) yields a true metric, (ii) is computable from finite samples, which is crucial since the distributions α_y are not available in analytic form, and (iii) is able to deal with sparsely-supported distributions.

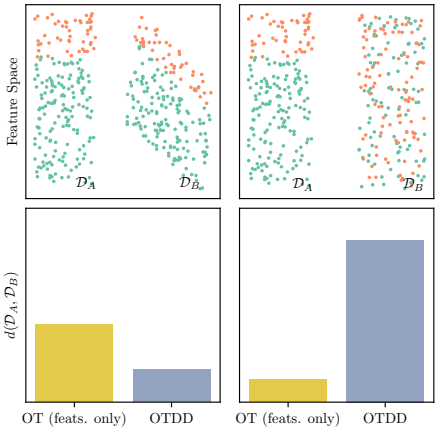


Figure 2: The importance of labels. The second pair of datasets are much closer than the first under the usual (label-agnostic) OT distance, while the opposite is true for our (label-aware) distance.

The approach described so far *grounds* the comparison of the α_y distributions to the feature space $\mathcal{X}$, so we can simply use $d_{\mathcal{X}}^p$ as the optimal transport cost, leading to a p-Wasserstein distance between labels: $\mathrm{W}_p^p(\alpha_y, \alpha_{y'})$, and in turn, to the following distance between feature-label pairs:

$$d_{\mathcal{Z}}\big((x, y), (x', y')\big) \triangleq \big(d_{\mathcal{X}}(x, x')^p + \mathrm{W}_p^p(\alpha_y, \alpha_{y'})\big)^{1/p}. \tag{5}$$

With this notion of distance in $\mathcal{Z}$ at hand, we can finally use optimal transport again to lift this point-wise metric into a distance between measures (and therefore, between datasets):

$$d_{\text{OT}}(\mathcal{D}_A, \mathcal{D}_B) = \min_{\pi \in \Pi(\alpha,\beta)} \int_{\mathcal{Z}\times\mathcal{Z}} d_{\mathcal{Z}}(z, z')^p \pi(z, z'). \tag{6}$$

As proved in Appendix A, this defines a true metric on $\mathcal{P}(\mathcal{Z})$ – the Optimal Transport Dataset Distance (OTDD). Figure 1 illustrates the main aspects of this distance on a simple 2D dataset.

It remains to describe how the distributions α_y are to be represented. One could treat the samples in $\mathsf{N}_{\mathcal{D}}(y)$ as support points of a uniform empirical measure, *i. e.,* $\alpha_y = \sum_{\mathbf{x}^{(i)} \in \mathsf{N}_{\mathcal{D}}(y)} \frac{1}{n_y} \delta_{\mathbf{x}^{(i)}}$, as described in Section 3. In this case, *every* evaluation of (5) would involve solving an OT problem, for a total worst-case $O(n^5 \log n)$ complexity, as shown in §C.1. This might be prohibitive in some settings. For those cases, we propose an alternative approach that relies on representing the α_y as Gaussian distributions, which leads to a simple yet tractable realization of the distance (6). Formally, we model each α_y as a Gaussian $\mathcal{N}(\hat{\mu}_y, \hat{\Sigma}_y)$ whose parameters are the sample mean and covariance of $\mathsf{N}_{\mathcal{D}}(y)$. The main advantage of this approach is that the 2-Wasserstein distance between Gaussians $\mathcal{N}(\mu_\alpha, \Sigma_\alpha)$ and $\mathcal{N}(\mu_\beta, \Sigma_\beta)$ has an analytic form, often known as the Bures-Wasserstein distance:

$$\mathrm{W}_2^2(\alpha, \beta) = \|\mu_\alpha - \mu_\beta\|_2^2 + \operatorname{tr}(\Sigma_\alpha + \Sigma_\beta - 2(\Sigma_\alpha^{\frac{1}{2}} \Sigma_\beta \Sigma_\alpha^{\frac{1}{2}})^{\frac{1}{2}}) \tag{7}$$

where $\Sigma^{\frac{1}{2}}$ is the matrix square root. Furthermore, whenever Σ_α and Σ_β commute, this further simplifies to

$$\mathrm{W}_2^2(\alpha, \beta) = \|\mu_\alpha - \mu_\beta\|_2^2 + \|\Sigma_\alpha^{\frac{1}{2}} - \Sigma_\beta^{\frac{1}{2}}\|_F^2. \tag{8}$$

When using the Bures-Wasserstein distance in $d_{\mathcal{Z}}$ (5), we denote the resulting dataset distance (6) by $d_{\text{OT-}\mathcal{N}}$, or Bures-OTDD.

Representing label-induced distributions as Gaussians might seem like a heuristic —and potentially, overly coarse— approximation. In cases where the data is first embedded with some complex non-linear mapping (*e. g.,* a neural network, as in our text classification experiments §6.4), there is empirical evidence that the first two moments capture enough relevant information for classification [23]. On the other hand, the following result, a consequence of a bound by Gelbrich [26], shows that we can provably 'sandwich' the exact d_{OT} by this Gaussian approximation and a trivial and easily computable upper bound (defined in Appendix B):

Proposition 4.1. *For any two datasets $\mathcal{D}_A, \mathcal{D}_B$, we have:*

$$d_{\text{OT-}\mathcal{N}}(\mathcal{D}_A, \mathcal{D}_B) \leq d_{\text{OT}}(\mathcal{D}_A, \mathcal{D}_B) \leq d_{\text{UB}}(\mathcal{D}_A, \mathcal{D}_B) \tag{9}$$

where d_{UB} is a distribution-agnostic OT upper bound. Furthermore, the first two distances are equal if all the label distributions α_y are Gaussian or elliptical (i. e., $d_{\text{OT-}\mathcal{N}}$ is exact in that case).

In the next section, we compare d_{OT} and $d_{\text{OT-}\mathcal{N}}$ in terms of their computational complexity, and in Section 6.1 we perform ablations comparing these and other baselines, including the lower and upper bounds of Proposition 4.1, in controlled experimental settings.

5 Computational Considerations

Since our goal in this work is to leverage dataset distances for tasks like transfer learning in realistic (*i. e.,* large) machine learning datasets, scalability is crucial. Indeed, most compelling use cases of *any* notion of distance between datasets will involve computing it repeatedly on very large samples. While estimation of Wasserstein —and more generally, optimal transport— distances is known to be computationally expensive in general, in Section 3 we mentioned how entropy regularization can be used to trade-off accuracy for runtime. Recall that both the general and Gaussian versions of the dataset distance proposed in Section 4 involve solving optimal transport problems (though the latter, owing to the closed-form solution of subproblem (7), only requires optimization for the global problem). Therefore, both of these distances benefit from approximate OT solvers.

But further speed-ups are possible. For $d_{\text{OT-}\mathcal{N}}$, a simple and fast implementation can be obtained if (i) the metric in $\mathcal{X}$ coincides with the ground metric in the transport problem on $\mathcal{Y}$, and (ii) all covariance matrices commute. While (ii) will rarely occur in practice, one could use a diagonal approximation to the covariance, or with milder assumptions, simultaneous matrix diagonalization [18]. In either case, using the simplification in (8), the pointwise distance $d(z, z')$ can be computed

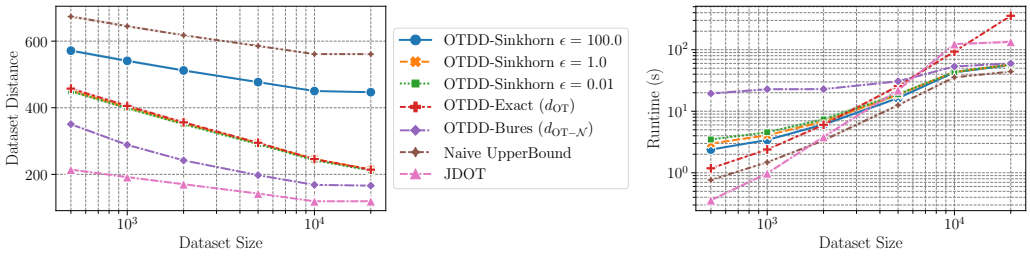


Figure 3: Comparison of variants of OTDD using different methods to compute the label distance $d_{\mathcal{Y}}$, and other baselines. The distances are computed between subsets of MNIST drawn independently.

by creating augmented representations of each dataset, whereby each pair (x, y) is represented as a stacked vector $\tilde{x} := [x; \mu_y; \text{vec}(\Sigma_y^{1/2})]$ for the corresponding label mean and covariance. Then, $\|\tilde{x} - \tilde{x}'\|_2^2 = d_{\mathcal{Z}}(x, y; x', y')^2$ for $d_{\mathcal{Z}}$ as defined in Eq. (5). Therefore, in this case the OTDD can be immediately computed using an off-the-shelf OT solver on these augmented datasets. While this approach is appealing computationally, here instead we focus on a exact version that does not require diagonal or commuting covariance approximations, and leave empirical evaluation of this approximate approach for future work.

The steps we propose next are motivated by the observation that, unlike traditional OT distances for which the cost of computing pair-wise distance is negligible compared to the complexity of the optimization routine, in our case the latter dominates, since it involves computing multiple OT distances itself. In order to speed up computation, we first pre-compute and store in memory all label-to-label pairwise distances $d(\alpha_y, \alpha_{y'})$, and retrieve them on-demand during the optimization of the outer OT problem. For $d_{\text{OT-}\mathcal{N}}$, computing the label-to-label distances $d(\mathcal{N}(\hat{\mu}_y, \hat{\Sigma}_y), \mathcal{N}(\hat{\mu}_{y'}, \hat{\Sigma}_{y'}))$ is dominated by the cost of computing matrix square roots, which if done exactly involves a full eigendecomposition. Instead, it can be computed approximately using the Newton-Schulz iterative method [29, 44]. Besides runtime, loading all examples of a given class to memory (to compute means and covariances) might be infeasible for large datasets (especially if running on GPU), so we instead use a two-pass stable online batch algorithm to compute these statistics [10].

The following result, proven in the Supplement §C, summarizes the time complexity of our two distances and sheds light on the trade-off between precision and efficiency they provide.

Theorem 5.1. *For datasets of size n and m, with p and q classes, dimension d, and maximum class size $\mathfrak{n}$, both d_{OT} and $d_{\text{OT-}\mathcal{N}}$ incur in a cost of $O(nm \log(\max\{n, m\})\tau^{-3})$ for solving the outer OT problem τ-approximately, while the worst-case complexity for computing the label-to-label pairwise distances (5) is $O\big(nm(d + \mathfrak{n}^3 \log \mathfrak{n} + d\mathfrak{n}^2)\big)$ for d_{OT} and $O\big(nmd + pqd^3 + d^2\mathfrak{n}(p + q)\big)$ for $d_{\text{OT-}\mathcal{N}}$.*

For small to medium-sized datasets, computing the exact d_{OT} is feasible and might even be faster than computing $d_{\text{OT}-\mathcal{N}}$, *e. g.,* when $d \gg n$, in which case d^3 dominates the complexity in Theorem 5.1. This can be observed in practice too (Fig. 3). For very large datasets ($n \gg d$), the cost of computing pairwise distances will often dominate. For example, if $n = m$ and the largest class size is $O(n)$, this step becomes $O(n^5 \log n)$ —prohibitive for large datasets— for d_{OT} but only $O(n^2 d + d^3)$ for $d_{\text{OT-}\mathcal{N}}$. In such cases, $d_{\text{OT-}\mathcal{N}}$ might be the only viable option.

6 Experiments

6.1 Asymptotics and Runtime for Variations of the OTDD

In our first set of experiments, we investigate the behavior and quality of the dataset distance proposed here for variations on how the label-to-label distance $d_{\mathcal{Y}}$ is computed. Recall that the two main approaches do so proposed here are to compute it as an exact distance between empirical measures (leading to d_{OT}) or using a Gaussian approximation ($d_{\text{OT-}\mathcal{N}}$). For the former, we discussed in Section 5 possible speed-ups by solving this inner OT problem approximately using a Sinkhorn divergence (4). We compare these three variants (exact, bures, sinkhorn), along with three other baselines: the upper bound of Theorem 5.1, the means-only approximation of the label distributions discussed in Section 4, and the JDOT approach [14] that uses a classification loss. Note that this last approach is

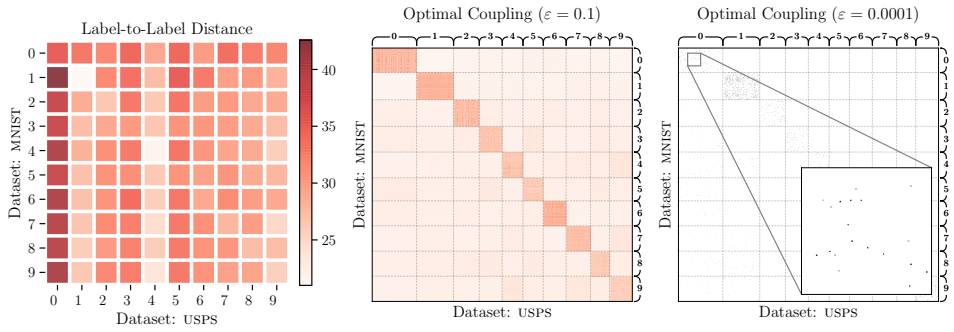


Figure 4: Dataset Distance between MNIST and USPS. **Left**: The label Wasserstein distances — computed without knowledge of the relation between labels across domains— recover expected relations between these classes. **Center/Right**: The optimal coupling π^* for different regularization levels exhibits a block-diagonal structure, indicating class-coherent matches across domains.

not directly comparable, as its scaling depends on the classification loss used, and is only applicable to settings where the two datasets have identical label sets. For our experiments, we take independent samples of MNIST of increasing size, and compute distance with all these methods. The results (Figure 3) exhibit various interesting phenomena. First, consistent with Proposition 4.1, the exact d_{OT} is bounded by d_{UB} and the approximate $d_{\text{OT-}\mathcal{N}}$, with the latter being remarkably tight, particularly as dataset size grows. The Sinkhorn-based versions of OTDD interpolate between these two bounds. Finally, as predicted by Theorem 5.1, in this case the exact OTDD is in fact faster to compute that the approximate $d_{\text{OT-}\mathcal{N}}$ for small dataset sizes ($\lesssim$5K samples), and again, using Sinkhorn for the inner OT problem allows us to interpolate between these two regimes.

6.2 Dataset Selection for Transfer Learning

In this section, we test whether the OTDD can provide a learning-free criterion on which to select a source dataset for transfer learning. We start with a simple domain adaptation setting, using USPS, MNIST [36] and three of its extensions: Fashion-MNIST [54], KMNIST [11] and the `letters` split of EMNIST [12]. All datasets consist of 10 classes, except EMNIST, for which the selected split has 26 classes. Throughout this section, we use a simple LeNet-5 neural network (two convolutional layers, three fully connected ones) with ReLU activations. When carrying out adaptation, we freeze the convolutional layers and fine-tune only the top three layers.

We first compute all pairwise OTDD distances (Fig 5). For the example of $d_{\text{OT-}\mathcal{N}}(\text{MNIST}, \text{USPS})$, Figure 4 illustrates two key components of the distance: the label-to-label distances $d_{\mathcal{Y}}$ (left) and the optimal coupling π^* obtained for two levels of entropy regularization ε (center, right). The diagonal elements of the first plot (*i. e.,* distances between congruent digit classes) are overall relatively smaller than off-diagonal elements. Interestingly, the `0` class of USPS appears remarkably far from *all* MNIST digits under this metric. On the other hand, most correspondences lie along the (block) diagonal of π^*, which shows the dataset distance is able to infer class-coherent correspondences across them. Despite both consisting of digits, MNIST and USPS are not the closest among these datasets according to the OTDD, as Figure 5 shows. The closest pair is instead (MNIST, EMNIST), while Fashion-MNIST appears far from all others, particularly MNIST.

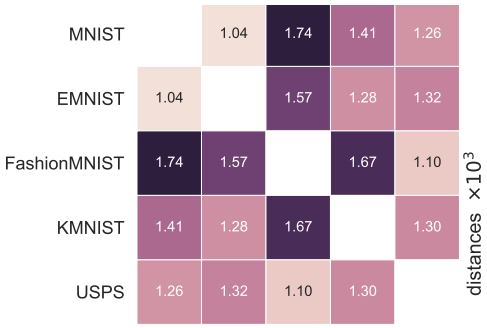
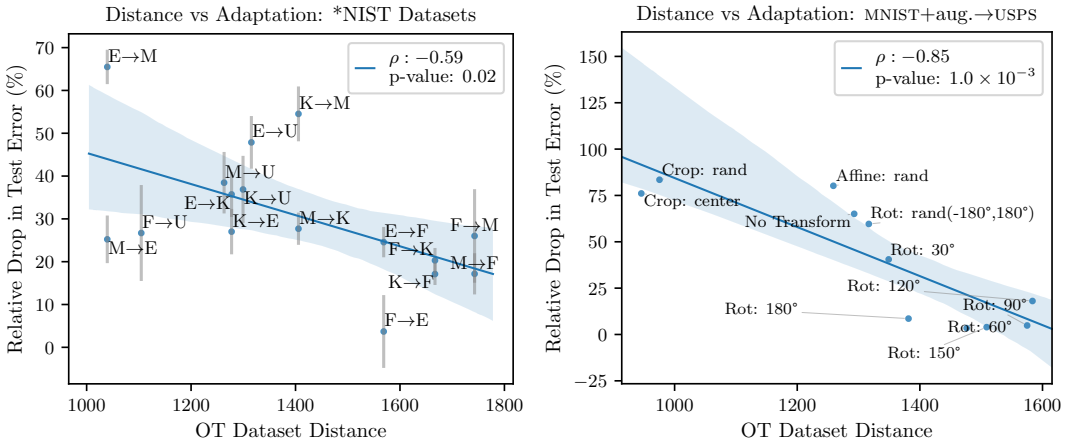


Figure 5: Pairwise OT Distances

We test the robustness of the distance by computing it repeatedly for varying sample sizes. The results (Fig. 9, Appendix G) show that the distance converges towards a fixed value as sample sizes grow, but interestingly, small sample sizes for USPS lead to wider variability, suggesting that this dataset itself is more heterogeneous than MNIST.



(a) All pairs of *NIST & USPS datasets

(b) MNIST (+augmentations)→USPS

Figure 6: Comparison of the OT dataset distance and transferability between *NIST datasets.

Next, we compare the OTDD against the *transferability* between datasets, *i. e.,* the gain in performance from using a model pretrained on the source domain and fine-tuning it on the target domain, compared to not pretraining. To make these numbers comparable across dataset pairs, we report *relative* drop in classification error brought by adaptation: $\mathcal{T}(\mathcal{D}_S \to \mathcal{D}_T) = 100 \times \frac{\text{error}(\mathcal{D}_S \to \mathcal{D}_T) - \text{error}(\mathcal{D}_T)}{\text{error}(\mathcal{D}_T)}$.

We run the adaptation task 10 times with different random seeds for each pair of datasets, and compare $\mathcal{T}$ against their distance. The strong and significant correlation between these (Fig. 6a) shows that the OTDD is highly predictive of transferability across these datasets. In particular, EMNIST led to the best adaptation to MNIST, justifying the —initially counter-intuitive— value of the OTDD. Repeating this experiment with ablated versions of the OTDD shows that using both feature and label information, and modeling second-order moments, are crucial to achieve this strength of correlation with transferability (Figure 8, Appendix F).

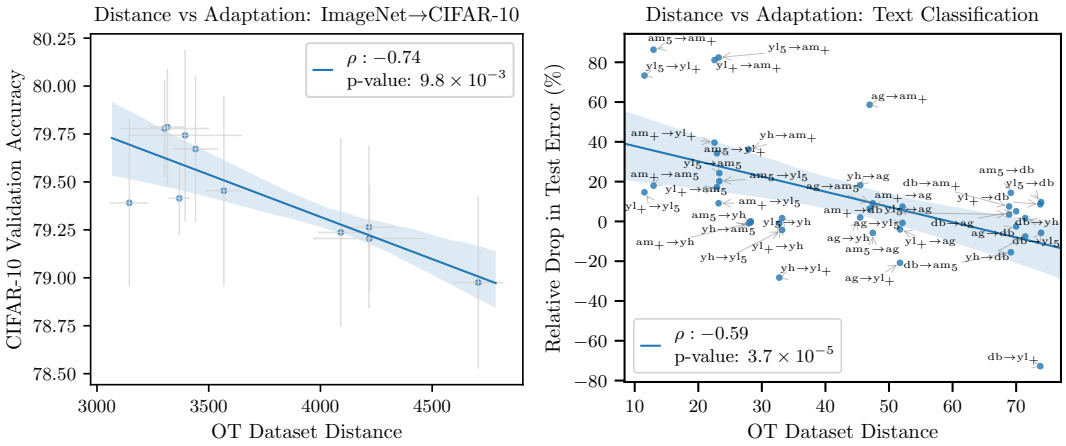
6.3 Distance-Driven Data Augmentation

Data augmentation —*i. e.,* applying carefully chosen transformations on a dataset to enhance its quality and diversity— is another key aspect of transfer learning that has substantial empirical effect on the quality of the transferred model yet lacks principled guidelines. Here, we investigate if the OTDD could be used to compare and select among possible augmentations.

For a fixed source-target dataset pair, we generate copies of the source data with various transformations applied to it, compute their distance to the target dataset, and compare to transfer accuracy as before. We present results for a small-scale (MNIST→USPS) and a larger-scale (Tiny-ImageNet→CIFAR-10) setting. The transformations we use on MNIST consist of rotations by a fixed degree $[30°, \dots, 180°]$, random rotations $(-180°, 180°)$, random affine transformations, center- and random-crops. For Tiny-ImageNet we randomly vary brightness, contrast, hue and saturation. The models used are respectively the LeNet-5 and a ResNet-50 (training details in Appendix §E). The results in both of these settings (Figures 6b and 7a) show, again, a strong significant correlation between these two. A reader familiar with the MNIST and USPS datasets will not be surprised by the fact that cropping images from the former leads to substantially better performance on the latter, while most rotations degrade transferability.

6.4 Transfer Learning for Text Classification

Natural Language Processing (NLP) has recently seen a profound impact from large-scale transfer learning, largely driven by the availability of off-the-shelf large language models pre-trained on massive amounts of data [21, 45, 47]. While natural language inherently lacks the fixed-size continuous vector representation required by our framework to compute pointwise distances, we can take advantage of these pretrained models to embed sentences in a vector space with rich geometry. In our experiments, we first embed the sentences of every dataset using the (base) BERT model [21], and then compute the OTDD on these embedded datasets.



(a) Tiny-ImageNet(+augmentations)→CIFAR-10 **(b)** Text datasets (see key below) embedded w/BERT

Figure 7: Comparison of the OT dataset distance and transferability in two large-scale settings.

We focus on the problem of sentence classification, and consider the following datasets[2] by Zhang et al. [56]: AG news (ag), DBPedia (db), Yelp Reviews with 5-way classification (yl_5) and binary polarity (yl_+) label encodings, Amazon Reviews with 5-way classification (am_5) and binary polarity (am_+) label encodings, and YAHOO Answers (yh). We provide details for all these datasets in Appendix D.

As before, we simulate a challenging adaptation setting by keeping only 100 examples per target class. For every pair of datasets, we first fine-tune the BERT model using the entirety of the source domain data, after which we fine-tune and evaluate on the target domain. Figure 7b shows that the OT dataset distance is highly correlated with transferability in this setting too. Interestingly, adaptation often leads to drastic degradation of performance in this case, which suggests that off-the-shelf BERT is on its own powerful and flexible enough to initialize many of these tasks, and therefore choosing the wrong domain for initial training might destroy some of that information.

7 Discussion

We have shown that the notion of distance between datasets proposed in this work is scalable and flexible, all the while offering appealing theoretical properties and interpretable comparisons. To allow for scaling up to very large datasets, we have proposed approximate versions of the OTDD, which despite trading off quality for runtime, often perform almost as well as the exact one in practice. Naturally, any refinement over these approximations would only further mitigate this trade-off. In terms of applications, our results on transfer learning scenarios demonstrate the practical utility of this novel distance, and are consistent with prior work showing that adaptation is most likely —and often *only*— successful if the domains are not too different [41].

There are many natural extensions and varitions of this distance. Here we assumed that the datasets were defined on feature spaces of the same dimension, but one could instead leverage a relational notion such as the Gromov-Wasserstein distance [43] to compute the distance between datasets whose features and not directly comparable. On the other hand, our efficient implementation relies on modeling groups of points with the same label as Gaussian distributions. This could naturally be extended to more general distributions for which the Wasserstein distance either has an analytic solution or at least can be computed efficiently, such as elliptic distributions [44], Gaussian mixture models [19], certain Gaussian Processes [40], or tree metrics [35].

In this work, we purposely excluded two key aspects of any learning task from our notion of distance: the loss function and the predictor function class. While we posit that it is crucial to have a notion of distance that is independent of these choices, it is nevertheless appealing to ask whether our distance could be extended to take those into account, ideally involving minimal training. Exploring different avenues to inject such information into this framework will be the focus of our future work.

[2] Available via the `torchtext` library.

Broader Impact

A notion of distance is such a basic and fundamental concept that it is most often used as a primitive from which other tools and methods derive utility. In the specific case of the dataset distance we propose here, it would most likely be used as tool within a machine learning pipeline. Thus, by its very nature, the prospect of potential impact of this work is broad enough to essentially encompass most settings where machine learning is used. In this statement, we focus on aspects that are immediate, tractable, and precise enough to be discussed constructively in this format.

Perhaps the most immediate impact of this work could be through its application in transfer learning. Improvements in this paradigm can have a myriad outcomes, ranging from societal to environmental, both within and beyond the machine learning community. Among potential beneficial outcomes, one that stands out is the environmental impact of making transfer learning more efficient by providing guidance as to what resources to use for pretraining (§6.2) or choosing optimal data augmentations (§6.3). This would be particularly relevant for NLP, where the carbon footprint of models has grown exponentially in recent years, driven largely by pretraining of very large models on massive datasets [49]. Another beneficial outcome of this specific use of the distance proposed in this work rests on the intuition that more efficient transfer learning would could erode or mitigate economic barriers that currently limit large-scale data pretraining and adaptation to resource-rich entities and institutions. However, work studying the impact of improved data and method efficiency has pointed out that this intuition is perhaps too optimistic, as there are various unexpected yet feasible negative collateral consequences of increased efficiency, *e. g.,* in terms of privacy, data markets and misuse [51].

We next highlight a few potential failure modes of this work. The modeling approximations used here to make this notion of distance efficiently computable, in particular the use of Gaussian distribution for modeling same-class collections, might prove too unrealistic in some datasets, leading to unreliable distance estimation. This, of course, could have negative impact on downstream applications that would rely on this distance as a sub-component, especially so given how deeply embedded within an ML pipeline it would be. In order to mitigate such impact, we suggest the practitioner verify how realistic these modeling assumptions are for the application at hand. On the other hand, despite the limited number of hyperparameters the computation of this distance relies on, inadequate choices for these (*e. g.,* the entropy regularization parameter ε) might nevertheless lead to unreliable or imprecise results. Again, care should be taken in test the validity of the parameters, ideally running sanity-checks on identical or near-identical datasets to corroborate that the results are sensible.

Acknowledgments and Disclosure of Funding

D.A.M. and N.F. were employed by Microsoft corporation while performing this work.

References

[1] A. Achille et al. "Dynamics and Reachability of Learning Tasks". In: (Oct. 2018). arXiv: 1810.02440 [cs.LG].

[2] A. Achille et al. "Task2Vec: Task Embedding for Meta-Learning". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2019, pp. 6430–6439.

[3] J. Altschuler et al. "Near-linear time approximation algorithms for optimal transport via Sinkhorn iteration". In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon et al. Curran Associates, Inc., 2017, pp. 1964–1974.

[4] D. Alvarez-Melis and T. Jaakkola. "Gromov-Wasserstein Alignment of Word Embedding Spaces". In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 2018, pp. 1881–1890.

[5] D. Alvarez-Melis et al. "Structured Optimal Transport". In: *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*. Ed. by Amos Storkey And. Vol. 84. Proceedings of Machine Learning Research. PMLR, 2018, pp. 1771–1780.

[6] S.-I. Amari. *Differential-Geometrical Methods in Statistics*. Vol. 28. Lecture Notes in Statistics. New York, NY: Springer New York, 1985.

[7] S.-I. Amari. "Natural Gradient Works Efficiently in Learning". In: *Neural Comput.* 10.2 (Feb. 1998), pp. 251–276.

[8] S.-I. Amari and H. Nagaoka. *Methods of Information Geometry*. en. Translations of Mathematical Monographs. American Mathematical Society, 2000.

[9] S. Ben-David et al. “Analysis of Representations for Domain Adaptation”. In: *Advances in Neural Information Processing Systems 19*. Ed. by B. Schölkopf et al. MIT Press, 2007, pp. 137–144.

[10] T. F. Chan et al. “Algorithms for Computing the Sample Variance: Analysis and Recommendations”. In: *Am. Stat.* 37.3 (Aug. 1983), pp. 242–247.

[11] T. Clanuwat et al. “Deep Learning for Classical Japanese Literature”. In: (Dec. 2018). arXiv: 1812.01718 [cs.CV].

[12] G. Cohen et al. “EMNIST: Extending MNIST to handwritten letters”. In: *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE, May 2017, pp. 2921–2926.

[13] C. Cortes and M. Mohri. “Domain Adaptation in Regression”. In: *Algorithmic Learning Theory*. Springer Berlin Heidelberg, 2011, pp. 308–323.

[14] N. Courty et al. “Joint distribution optimal transportation for domain adaptation”. In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon et al. Curran Associates, Inc., 2017, pp. 3730–3739.

[15] N. Courty et al. “Optimal Transport for Domain Adaptation”. en. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 39.9 (Sept. 2017), pp. 1853–1865.

[16] M. Cuturi. “Sinkhorn Distances: Lightspeed Computation of Optimal Transport”. In: *Advances in Neural Information Processing Systems 26*. Ed. by C. J. C. Burges et al. Curran Associates, Inc., 2013, pp. 2292–2300.

[17] B. B. Damodaran et al. “DeepJDOT: Deep Joint Distribution Optimal Transport for Unsupervised Domain Adaptation”. In: *Computer Vision – ECCV 2018*. Springer International Publishing, 2018, pp. 467–483.

[18] L. De Lathauwer. “Simultaneous matrix diagonalization: the overcomplete case”. In: *Proc. of the 4th International Symposium on ICA and Blind Signal Separation, Nara, Japan*. Vol. 8122. kecl.ntt.co.jp, 2003, p. 825.

[19] J. Delon and A. Desolneux. “A Wasserstein-type distance in the space of Gaussian Mixture Models”. In: (July 2019). arXiv: 1907.05254 [math.OC].

[20] J. Deng et al. “ImageNet: A large-scale hierarchical image database”. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, June 2009, pp. 248–255.

[21] J. Devlin et al. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. 2019, pp. 4171–4186.

[22] Y. Dukler et al. “Wasserstein of Wasserstein Loss for Learning Generative Models”. In: *Proceedings of the 36th International Conference on Machine Learning*. Ed. by K. Chaudhuri and R. Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. Long Beach, California, USA: PMLR, 2019, pp. 1716–1725.

[23] M. El Amine Seddik et al. “Random Matrix Theory Proves that Deep Learning Representations of GAN-data Behave as Gaussian Mixtures”. In: *International Conference on Machine Learning*. Vienna, Austria: PMLR, 2020.

[24] J. Feydy et al. “Interpolating between Optimal Transport and MMD using Sinkhorn Divergences”. In: *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*. Ed. by K. Chaudhuri and M. Sugiyama. Vol. 89. Proceedings of Machine Learning Research. PMLR, 2019, pp. 2681–2690.

[25] C. Frogner et al. “Learning Embeddings into Entropic Wasserstein Spaces”. In: *International Conference on Learning Representations*. May 2019.

[26] M. Gelbrich. “On a Formula for the L2 Wasserstein Metric between Measures on Euclidean and Hilbert Spaces”. In: *Math. Nachr.* Lecture Notes in Control and Information Sciences 96 147.1 (Nov. 1990), pp. 185–203.

[27] A. Genevay et al. “Learning Generative Models with Sinkhorn Divergences”. In: *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*. Ed. by A. Storkey and F. Perez-Cruz. Vol. 84. Proceedings of Machine Learning Research. Playa Blanca, Lanzarote, Canary Islands: PMLR, 2018, pp. 1608–1617.

[28] A. Genevay et al. “Sample Complexity of Sinkhorn Divergences”. In: *Proceedings of Machine Learning Research*. Ed. by K. Chaudhuri and M. Sugiyama. Vol. 89. Proceedings of Machine Learning Research. PMLR, 2019, pp. 1574–1583.

[29] N. J. Higham. *Functions of Matrices: Theory and Computation*. en. SIAM, Jan. 2008.

[30] J. J. Hull. “A database for handwritten text recognition research”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 16.5 (May 1994), pp. 550–554.

[31] L. Kantorovitch. “On the Translocation of Masses”. In: *Dokl. Akad. Nauk SSSR* 37.7-8 (1942), pp. 227–229.

[32] M. Khodak et al. “Adaptive Gradient-Based Meta-Learning Methods”. In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach et al. Curran Associates, Inc., 2019, pp. 5915–5926.

[33] A. Krizhevsky and G. Hinton. “Learning multiple layers of features from tiny images”. 2009.

[34] D. Kuhn et al. “Wasserstein Distributionally Robust Optimization: Theory and Applications in Machine Learning”. In: (Aug. 2019). arXiv: 1908.08729 [stat.ML].

[35] T. Le et al. “Tree-Sliced Variants of Wasserstein Distances”. In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach et al. Curran Associates, Inc., 2019, pp. 12283–12294.

[36] Y. LeCun et al. “MNIST handwritten digit database”. In: (2010).

[37] R. Leite and P. Brazdil. “Predicting relative performance of classifiers from samples”. In: *Proceedings of the 22nd international conference on Machine learning*. dl.acm.org, 2005, pp. 497–503.

[38] L. Li. “Data Complexity in Machine Learning and Novel Classification Algorithms”. PhD thesis. California Institute of Technology, 2006.

[39] T. Liang et al. “Fisher-Rao Metric, Geometry, and Complexity of Neural Networks”. In: *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*. PMLR, 2019.

[40] A. Mallasto and A. Feragen. “Learning from uncertain curves: The 2-Wasserstein metric for Gaussian processes”. In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon et al. Curran Associates, Inc., 2017, pp. 5660–5670.

[41] Y. Mansour et al. “Domain Adaptation: Learning Bounds and Algorithms”. In: *The 22nd Conference on Learning Theory*. arxiv.org, 2009.

[42] F. Mémoli. “Distances Between Datasets”. In: *Modern Approaches to Discrete Curvature*. Ed. by L. Najman and P. Romon. Cham: Springer International Publishing, 2017, pp. 115–132.

[43] F. Mémoli. “Gromov–Wasserstein Distances and the Metric Approach to Object Matching”. In: *Found. Comput. Math.* 11.4 (Aug. 2011), pp. 417–487.

[44] B. Muzellec and M. Cuturi. “Generalizing Point Embeddings using the Wasserstein Space of Elliptical Distributions”. In: *Advances in Neural Information Processing Systems 31*. Ed. by S. Bengio et al. Curran Associates, Inc., 2018, pp. 10237–10248.

[45] M. Peters et al. “Deep Contextualized Word Representations”. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. aclweb.org, 2018, pp. 2227–2237.

[46] G. Peyré and M. Cuturi. “Computational Optimal Transport”. In: *Foundations and Trends® in Machine Learning* 11.5-6 (2019), pp. 355–607.

[47] A. Radford et al. “Better language models and their implications”. In: *OpenAI Blog https://openai.com/blog/better-language-models* (2019).

[48] Y. Rubner et al. “The Earth Mover’s Distance as a Metric for Image Retrieval”. In: *Int. J. Comput. Vis.* 40.2 (Nov. 2000), pp. 99–121.

[49] E. Strubell et al. “Energy and Policy Considerations for Deep Learning in NLP”. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. aclweb.org, 2019, pp. 3645–3650.

[50] A. T. Tran et al. “Transferability and Hardness of Supervised Classification Tasks”. In: (Aug. 2019). arXiv: 1908.08142 [cs.LG].

[51] A. D. Tucker et al. “Social and Governance Implications of Improved Data Efficiency”. In: *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*. AIES ’20. New York, NY, USA: Association for Computing Machinery, Feb. 2020, pp. 378–384.

[52] C. Villani. *Optimal transport, Old and New*. Vol. 338. Springer Science & Business Media, 2008.

[53] C. Villani. *Topics in Optimal Transportation*. en. American Mathematical Soc., 2003.

[54] H. Xiao et al. “Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms”. In: (Aug. 2017). arXiv: 1708.07747 [cs.LG].

[55] M. Yurochkin et al. “Hierarchical Optimal Transport for Document Representation”. In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach et al. Curran Associates, Inc., 2019, pp. 1599–1609.

[56] X. Zhang et al. “Character-level Convolutional Networks for Text Classification”. In: *Advances in Neural Information Processing Systems 28*. Ed. by C. Cortes et al. Curran Associates, Inc., 2015, pp. 649–657.

A OTDD is a True Distance

Proposition A.1. *$d_{\text{OT}}(\mathcal{D}_A, \mathcal{D}_B)$ defines a valid metric on $\mathcal{P}(\mathcal{X} \times \mathcal{P}(\mathcal{X}))$ the space of measures over feature and label-distribution pairs.*

Proof. Whenever the cost function used is a metric in a given space $\mathcal{X}$, the optimal transport problem itself defines a distance (the Wasserstein distance) on $\mathcal{P}(\mathcal{X})$ [52, Chapter 6]. Therefore, it suffices to show that the cost function $d_{\mathcal{Z}}$ defined in Eq. (5) is indeed a distance. Clearly, it is symmetric because both $d_{\mathcal{X}}$ and W_p are. In addition, since both of these are distances:

$$d_{\mathcal{Z}}(z, z') = 0 \Leftrightarrow d_{\mathcal{X}}(x, x') = 0 \wedge \text{W}_p(\alpha_y, \alpha'_y) = 0 \Leftrightarrow x = x',\ \alpha_y = \alpha'_y \Leftrightarrow z = z'$$

Finally, we have that

$$\begin{aligned} d_{\mathcal{Z}}(z_1, z_3) &= \left(d_{\mathcal{X}}(x_1, x_3)^p + \text{W}_p(\alpha_{y_1}, \alpha_{y_3})^p\right)^{\frac{1}{p}} \\ &\leq \left(d_{\mathcal{X}}(x_1, x_2)^p + d_{\mathcal{X}}(x_2, x_3)^p + \ \text{W}_p(\alpha_{y_1}, \alpha_{y_2})^p + \text{W}_p(\alpha_{y_2}, \alpha_{y_3})^p\right)^{\frac{1}{p}} \\ &= \left(d_{\mathcal{Z}}(z_1, z_2)^p + d_{\mathcal{Z}}(z_2, z_3)^p\right)^{\frac{1}{p}} = d_{\mathcal{Z}}(z_1, z_2) + d_{\mathcal{Z}}(z_2, z_3) \end{aligned}$$

where the last step is an application of Minkowski's inequality. Hence, $d_{\mathcal{Z}}$ satisfies the triangle inequality, and therefore it is a metric on $\mathcal{Z} = \mathcal{X} \times \mathcal{P}(\mathcal{X})$. We therefore conclude that the value of the optimal transport (6) that uses this metric as a cost function is a distance itself. □

B Proof of Proposition 4.1

Proposition 4.1 is a direct extension of the following well-known bound for the 2-Wasserstein distance due to Gelbrich [26]:

Lemma B.1 (Gelbrich bound)**.** *Suppose $\alpha, \beta \in \mathcal{P}(\mathbb{R}^d)$ are any two measures with mean vectors $\mu_\alpha, \mu_\beta \in \mathbb{R}^d$ and covariance matrices $\Sigma_\alpha, \Sigma_\beta \in \mathbb{S}^d_+$ respectively. Then,*

$$\text{W}_2^2\big(\mathcal{N}(\mu_\alpha, \Sigma_\alpha), \mathcal{N}(\mu_\beta, \Sigma_\beta)\big) \leq \text{W}_2^2(\alpha, \beta) \tag{10}$$

where $\text{W}_2^2\big(\mathcal{N}(\mu_\alpha, \Sigma_\alpha), \mathcal{N}(\mu_\beta, \Sigma_\beta)\big)$ is as in Eq. (7).

Obtaining an upper bound is trivial, noting that for any two measures α, β,

$$\text{W}_2^2(\alpha, \beta) = \|\mu_\alpha - \mu_\beta\|_2^2 + \text{tr}(\Sigma_\alpha + \Sigma_\beta) - 2 \max_{\pi \in \Pi} \text{tr}(\Sigma_\pi) \leq \|\mu_\alpha - \mu_\beta\|_2^2 + \text{tr}(\Sigma_\alpha + \Sigma_\beta). \tag{11}$$

Let $d_{\text{UB}}(\mathcal{D}_A, \mathcal{D}_B)$ denote the OT distance obtained by using the cost $d^2_{\mathcal{Z}}(z, z') = d_{\mathcal{X}}(x, x')^2 + \|\mu_y - \mu_{y'}\|_2^2 + \text{tr}(\Sigma_y + \Sigma_{y'})$. Then, for our setting, we have:

Proposition 4.1. *For any two datasets $\mathcal{D}_A, \mathcal{D}_B$, we have:*

$$d_{\text{OT-}\mathcal{N}}(\mathcal{D}_A, \mathcal{D}_B) \leq d_{\text{OT}}(\mathcal{D}_A, \mathcal{D}_B) \leq d_{\text{UB}}(\mathcal{D}_A, \mathcal{D}_B) \tag{12}$$

where d_{UB} is a distribution-agnostic OT upper bound. Furthermore, the first two distances are equal if all the label distributions α_y are Gaussian or elliptical (i. e., $d_{\text{OT-}\mathcal{N}}$ is exact in that case).

Proof. In the notation of Section 3, Lemma B.1 implies that for every feature-label pairs $z = (x, y)$ and $z' = (x', y')$, we have:

$$d_{\mathcal{X}}(x, x')^2 + \text{W}_2^2\big(\mathcal{N}(\mu_y, \Sigma_y), \mathcal{N}(\mu_{y'}, \Sigma_{y'})\big) \leq d_{\mathcal{X}}(x, x')^2 + \text{W}_2^2(\alpha_y, \alpha_{y'}), \tag{13}$$

and therefore

$$\int d_{\mathcal{Z}}(z, z')^2 \, \text{d}\pi \leq \int d_{\mathcal{Z}}(z, z')^2 \, \text{d}\pi \tag{14}$$

for every coupling $\pi \in \Pi(\alpha, \beta)$. In particular, for the minimizing π^*, we obtain that

$$d_{\text{OT-}\mathcal{N}}(\mathcal{D}_A, \mathcal{D}_B) \leq d_{\text{OT}}(\mathcal{D}_A, \mathcal{D}_B) \tag{15}$$

We obtain the upper bound analogously.

Clearly, Gelbrich's bound holds with equality when α and β are indeed Gaussian. More generally, equality is attained for elliptical distributions with the same density generator [34]. This immediately implies equality of the first two terms in equation (15) in that case. □

C Time Complexity Analysis

For the analyses in this section, assume that $\mathcal{D}_S$ and $\mathcal{D}_T$ respectively have n and m labeled examples in $\mathbb{R}^d$ and k_s, k_t classes. In addition, let $\mathsf{N}_{\mathcal{D}}^S(i) := \{x \in \mathcal{X} \mid (x, y = i) \in \mathcal{D}\}$ be the subset of examples in $\mathcal{D}_S$ with label i, and define analogously $\mathsf{N}_{\mathcal{D}}^T(j)$. The denote the cardinalities of these subsets as $n_s^i \triangleq |\mathsf{N}_s^{(i)}|$ and analogously for n_t^j.

Direct computation of the distance (5) involves two main steps:

(i) computing pairwise pointwise distances (each requiring solution of a label-to-label OT sub-problem), and

(ii) a global OT problem between the two samples.

Step (ii) is identical for both the general distance d_{OT} and its Gaussian approximation counterpart $d_{\text{OT-}\mathcal{N}}$, so we analyze it first. This is an OT problem between two discrete distributions of size n and m, which can be solved exactly in $O\big((n+m)nm\log(nm)\big)$ using interior point methods or Orlin's algorithm for the uncapacitated min cost flow problem [46]. Alternatively, it can be solved τ-approximately in $O(nm\log(\max\{n,m\})\tau^{-3})$ time using the Sinkhorn algorithm [3].

We next analyze step (i) individually for the two OTDD versions. Combined, they provide a proof of Theorem 5.1.

C.1 Pointwise distance computation for d_{OT}

Consider a single pair of points, $(x, y = i) \in \mathcal{D}_A$ and $(x', y' = j) \in \mathcal{D}_B$. Evaluating $\|x - x'\|$ has $O(d)$ complexity, while $W(\alpha_y, \beta_{y'})$ is an $n_s^i \times n_t^j$ OT problem which itself requires computing a distance matrix (at cost $O(n_s^i n_t^j d)$), and then solving the OT problem, which as discussed before, be done exactly in $O\big((n_s^i + n_t^j)n_s^i n_t^j \log(n_s^i + n_t^j)\big)$ or τ-approximately in $O(n_s^i n_t^j \log(\max\{n_s^i, n_t^j\})\tau^{-3})$.

For simplicity, let us denote $\mathfrak{n}_s = \max_i n_s^i$, and $\mathfrak{n}_t = \max_j n_t^j$ the size of the largest label cluster in each dataset, and $\mathfrak{n} = \max\{\mathfrak{n}_s, \mathfrak{n}_t\}$ the overall largest one. Using these, and combining all of the above, the overall worst case complexity for the computation of the $n \times m$ pairwise distances can be expressed as

$$O\big(nm(d + \mathfrak{n}^3 \log \mathfrak{n} + d\mathfrak{n}^2)\big), \tag{16}$$

which is what we wanted to show.

□

C.2 Pointwise distance computation for $d_{\text{OT-}\mathcal{N}}$

As before, consider a pair of points $(x, y = i) \in \mathcal{D}_A$ and $(x', y' = j) \in \mathcal{D}_B$ whose cluster sizes are n_s^i and n_t^j respectively. As mentioned in Section 5, for $d_{\text{OT-}\mathcal{N}}$ we first compute all the per-class means and covariance matrices. This step is clearly dominated by latter, which is $O(d^2 n_s^i)$.[3] Considering all labels from both datasets, this amounts to a worst-case complexity of $O\big(d^2(k_s\mathfrak{n}_s + k_t\mathfrak{n}_t)\big)$.

Once the means and covariances have been computed, we precompute all the $k_s \times k_t$ pair-wise label-to-label distances $\mathrm{W}_2(\alpha_y, \beta_{y'})$ using Eq. (7). This computation is dominated by the matrix square roots. If done exactly, these involve a full eigendecomposition, at cost $O(d^3)$, so the total cost for this step is $O(k_s k_t d^3)$.

Finally, while computing the pairwise distance, we will incur in $O(nmd)$ to obtain $\|x - x'\|$. Putting all of these together, and replacing $\mathfrak{n}_s, \mathfrak{n}_t$ by $\mathfrak{n}$, we obtain a total cost for precomputing all the point-wise distances of:

$$O(nmd + k_s k_t d^3 + d^2\mathfrak{n}(k_s + k_t),$$

which concludes the proof.

□

[3] technically, this would be $O(d^\omega n_s^i)$ where ω is the coefficient of matrix multiplication, but we take $\omega = 3$ for simplicity.

D Dataset Details

Information about all the datasets used, including references, are provided in Table 1.

Dataset	Input Dimension	Number of Classes	Train Examples	Test Examples	Source
USPS	$16 \times 16^{*}$	10	7291	2007	[30]
MNIST	28×28	10	60K	10K	[36]
EMNIST (letters)	28×28	26	145K	10K	[12]
KMNIST	28×28	10	60K	10K	[11]
FASHION-MNIST	28×28	10	60K	10K	[54]
TINY-IMAGENET	$64 \times 64^{\ddagger}$	200	100K	10K	[20]
CIFAR-10	32×32	10	50K	10K	[33]
AG news	$768^{\dagger}$	4	120K	7.6K	[56]
DBPedia	$768^{\dagger}$	14	560K	70K	[56]
YELPREVIEW (Polarity)	$768^{\dagger}$	2	560K	38K	[56]
YELPREVIEW (Full Scale)	$768^{\dagger}$	5	650K	50K	[56]
AMAZONREVIEW (Polarity)	$768^{\dagger}$	2	3.6M	400K	[56]
AMAZONREVIEW (Full Scale)	$768^{\dagger}$	5	3M	650K	[56]
YAHOO ANSWERS	$768^{\dagger}$	10	1.4M	60K	[56]

Table 1: Summary of datasets used. $*$: we rescale the USPS digits to 28×28 for comparison to the *NIST datasets. ‡: we rescale Tiny-ImageNet to 32×32 for comparison to CIFAR-10. †: for text datasets, variable-length sentences are embedded to fixed-dimensional vectors using BERT.

E Optimization and Training Details

For the adaptation experiments on the *NIST datasets, we use a LeNet-5 architecture with ReLU non-linearities trained for 20 epochs using ADAM with learning rate 1×10^{-3}, weight decay 1×10^{-6}, and fine-tuned for 10 epochs on the target domain(s) using the same optimization parameters.

For the Tiny-ImageNet to CIFAR-10 adaptation results, we use a ResNet-50 trained for 300 epochs using SGD with learning rate 0.1 momentum 0.9 and weight decay 1×10^{-4} It was fine-tuned for 30 epochs on the target domain using SGD with same parameters except 0.01 learning rate. We discard pairs for which the variance on adaptation accuracy is beyond a certain threshold.

For the text classification experiments, we use a pretrained BERT architecture (the `bert-base-uncased` model of the `transformers`[4] library). We first embed all sentences using this model. Then, for each pair of source/target domains, we first fine-tune using ADAM with learning rate 2×10^{-5} for 10 epochs on the full source domain data, and the fine-tune on the restricted target domain data with the same optimization parameters for 2 epochs.

Our implementation of the OTDD relies on the `pot`[5] and `geomloss`[6] python packages.

F Ablation Experiments on Dataset Selection for Transfer Learning

We repeat the experimental setting of Section 6.2, now using three ablated versions of the OTDD: one which completely ignores the labels (*i. e.,* uses $d_{\mathcal{Z}} = d_{\mathcal{X}}$), one that completely ignores the features ($d_{\mathcal{Z}} = d_{\mathcal{Y}}$), and one that uses a means-only comparison of the label-induced distributions, that is, takes $d_{\mathcal{Y}}(y, y') = \|\mu_y - \mu_y\|$, which can be seen as using a first-order moment approximation of the Bures-Wasserstein distance. Comparing Figure 8 to Figure 6a, we see that both feature and label information is crucial for the OTDD to be predictive of transferability, although, interestingly, dropping the features is not as detrimental, probably because there is already substantial information about these encoded implicitly in the label distributions. On the other hand, the poor performance of the means-only distance shows that second order moment information is crucial.

[4] huggingface.co/transformers/

[5] pot.readthedocs.io/en/stable/

[6] www.kernel-operations.io/geomloss/

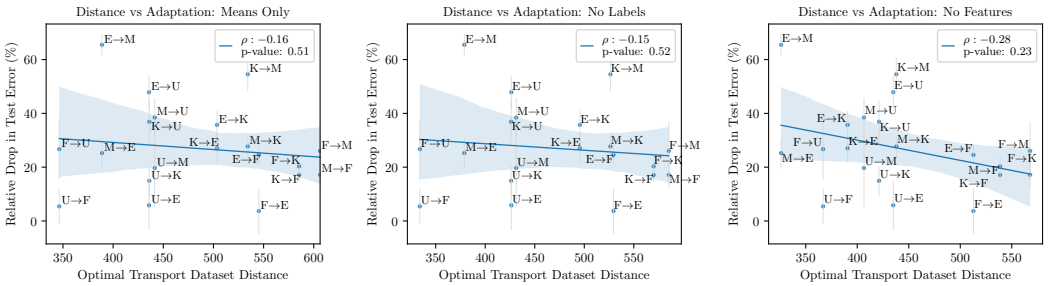


Figure 8: Comparison of ablated versions of OTDD for transferability prediction.

G Robustness of the Distance

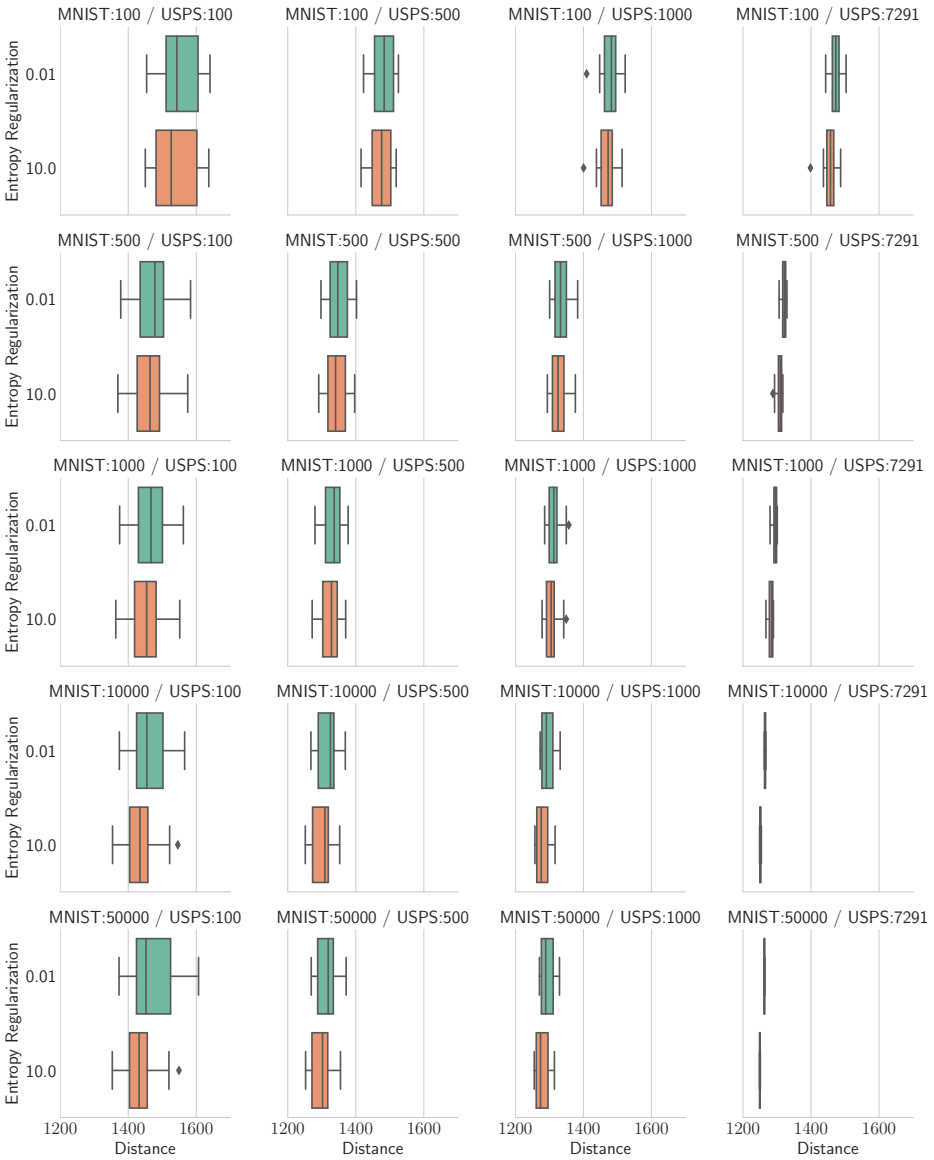


Figure 9: Robustness Analysis: distances computed on subsets of varying size (rows: MNIST, columns: USPS), over 10 random repetitions, for two values of the regularization parameter ε.