Learning with a Wasserstein Loss

Charlie Frogner* **Chiyuan Zhang***
Center for Brains, Minds and Machines
Massachusetts Institute of Technology
frogner@mit.edu, chiyuan@mit.edu

Hossein Mobahi
CSAIL
Massachusetts Institute of Technology
hmobahi@csail.mit.edu

Mauricio Araya-Polo
Shell International E & P, Inc.
Mauricio.Araya@shell.com

Tomaso Poggio
Center for Brains, Minds and Machines
Massachusetts Institute of Technology
tp@ai.mit.edu

Abstract

Learning to predict multi-label outputs is challenging, but in many problems there is a natural metric on the outputs that can be used to improve predictions. In this paper we develop a loss function for multi-label learning, based on the Wasserstein distance. The Wasserstein distance provides a natural notion of dissimilarity for probability measures. Although optimizing with respect to the exact Wasserstein distance is costly, recent work has described a regularized approximation that is efficiently computed. We describe an efficient learning algorithm based on this regularization, as well as a novel extension of the Wasserstein distance from probability measures to unnormalized measures. We also describe a statistical learning bound for the loss. The Wasserstein loss can encourage smoothness of the predictions with respect to a chosen metric on the output space. We demonstrate this property on a real-data tag prediction problem, using the Yahoo Flickr Creative Commons dataset, outperforming a baseline that doesn't use the metric.

1 Introduction

We consider the problem of learning to predict a non-negative measure over a finite set. This problem includes many common machine learning scenarios. In multiclass classification, for example, one often predicts a vector of scores or probabilities for the classes. And in semantic segmentation [1], one can model the segmentation as being the support of a measure defined over the pixel locations. Many problems in which the output of the learning machine is both non-negative and multi-dimensional might be cast as predicting a measure.

We specifically focus on problems in which the output space has a natural metric or similarity structure, which is known (or estimated) *a priori*. In practice, many learning problems have such structure. In the ImageNet Large Scale Visual Recognition Challenge [ILSVRC] [2], for example, the output dimensions correspond to 1000 object categories that have inherent semantic relationships, some of which are captured in the WordNet hierarchy that accompanies the categories. Similarly, in the keyword spotting task from the IARPA Babel speech recognition project, the outputs correspond to keywords that likewise have semantic relationships. In what follows, we will call the similarity structure on the label space the *ground metric* or *semantic similarity*.

Using the ground metric, we can measure prediction performance in a way that is sensitive to relationships between the different output dimensions. For example, confusing dogs with cats might

*Authors contributed equally.

[1]Code and data are available at http://cbcl.mit.edu/wasserstein.

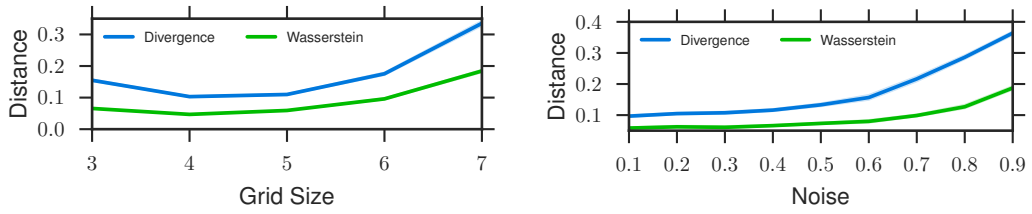


Figure 2: The Wasserstein loss encourages predictions that are similar to ground truth, robustly to incorrect labeling of similar classes (see Appendix E.1). Shown is Euclidean distance between prediction and ground truth vs. (left) number of classes, averaged over different noise levels and (right) noise level, averaged over number of classes. Baseline is the multiclass logistic loss.

be more severe an error than confusing breeds of dogs. A loss function that incorporates this metric might encourage the learning algorithm to favor predictions that are, if not completely accurate, at least semantically similar to the ground truth.

In this paper, we develop a loss function for multi-label learning that measures the *Wasserstein distance* between a prediction and the target label, with respect to a chosen metric on the output space. The Wasserstein distance is defined as the cost of the optimal transport plan for moving the mass in the predicted measure to match that in the target, and has been applied to a wide range of problems, including barycenter estimation [3], label propagation [4], and clustering [5]. To our knowledge, this paper represents the first use of the Wasserstein distance as a loss for supervised learning.

Siberian husky Eskimo dog

Figure 1: Semantically near-equivalent classes in ILSVRC

We briefly describe a case in which the Wasserstein loss improves learning performance. The setting is a multiclass classification problem in which label noise arises from confusion of semantically near-equivalent categories. Figure 1 shows such a case from the ILSVRC, in which the categories *Siberian husky* and *Eskimo dog* are nearly indistinguishable. We synthesize a toy version of this problem by identifying categories with points in the Euclidean plane and randomly switching the training labels to nearby classes. The Wasserstein loss yields predictions that are closer to the ground truth, robustly across all noise levels, as shown in Figure 2. The standard multiclass logistic loss is the baseline for comparison. Section E.1 in the Appendix describes the experiment in more detail.

The main contributions of this paper are as follows. We formulate the problem of learning with prior knowledge of the ground metric, and propose the Wasserstein loss as an alternative to traditional information divergence-based loss functions. Specifically, we focus on empirical risk minimization (ERM) with the Wasserstein loss, and describe an efficient learning algorithm based on entropic regularization of the optimal transport problem. We also describe a novel extension to unnormalized measures that is similarly efficient to compute. We then justify ERM with the Wasserstein loss by showing a statistical learning bound. Finally, we evaluate the proposed loss on both synthetic examples and a real-world image annotation problem, demonstrating benefits for incorporating an output metric into the loss.

2 Related work

Decomposable loss functions like KL Divergence and ℓ_p distances are very popular for probabilistic [1] or vector-valued [6] predictions, as each component can be evaluated independently, often leading to simple and efficient algorithms. The idea of exploiting smoothness in the label space according to a prior metric has been explored in many different forms, including regularization [7] and post-processing with graphical models [8]. Optimal transport provides a natural distance for probability distributions over metric spaces. In [3, 9], the optimal transport is used to formulate the Wasserstein barycenter as a probability distribution with minimum total Wasserstein distance to a set of given points on the probability simplex. [4] propagates histogram values on a graph by minimizing a Dirichlet energy induced by optimal transport. The Wasserstein distance is also used to formulate a metric for comparing clusters in [5], and is applied to image retrieval [10], contour

matching [11], and many other problems [12, 13]. However, to our knowledge, this is the first time it is used as a loss function in a discriminative learning framework. The closest work to this paper is a theoretical study [14] of an estimator that minimizes the optimal transport cost between the empirical distribution and the estimated distribution in the setting of statistical parameter estimation.

3 Learning with a Wasserstein loss

3.1 Problem setup and notation

We consider the problem of learning a map from $\mathcal{X} \subset \mathbb{R}^D$ into the space $\mathcal{Y} = \mathbb{R}_+^K$ of measures over a finite set $\mathcal{K}$ of size $|\mathcal{K}| = K$. Assume $\mathcal{K}$ possesses a metric $d_{\mathcal{K}}(\cdot, \cdot)$, which is called the *ground metric*. $d_{\mathcal{K}}$ measures semantic similarity between dimensions of the output, which correspond to the elements of $\mathcal{K}$. We perform learning over a hypothesis space $\mathcal{H}$ of predictors $h_\theta : \mathcal{X} \to \mathcal{Y}$, parameterized by $\theta \in \Theta$. These might be linear logistic regression models, for example.

In the standard statistical learning setting, we get an i.i.d. sequence of training examples $S = ((x_1, y_1), \ldots, (x_N, y_N))$, sampled from an unknown joint distribution $\mathcal{P}_{\mathcal{X}\times\mathcal{Y}}$. Given a measure of performance (a.k.a. *risk*) $\mathcal{E}(\cdot, \cdot)$, the goal is to find the predictor $h_\theta \in \mathcal{H}$ that minimizes the expected risk $\mathbb{E}[\mathcal{E}(h_\theta(x), y)]$. Typically $\mathcal{E}(\cdot, \cdot)$ is difficult to optimize directly and the joint distribution $\mathcal{P}_{\mathcal{X}\times\mathcal{Y}}$ is unknown, so learning is performed via *empirical risk minimization*. Specifically, we solve

$$\min_{h_\theta \in \mathcal{H}} \left\{ \hat{\mathbb{E}}_S[\ell(h_\theta(x), y) = \frac{1}{N} \sum_{i=1}^{N} \ell(h_\theta(x_i), y_i) \right\} \tag{1}$$

with a loss function $\ell(\cdot, \cdot)$ acting as a surrogate of $\mathcal{E}(\cdot, \cdot)$.

3.2 Optimal transport and the exact Wasserstein loss

Information divergence-based loss functions are widely used in learning with probability-valued outputs. Along with other popular measures like Hellinger distance and χ^2 distance, these divergences treat the output dimensions independently, ignoring any metric structure on $\mathcal{K}$.

Given a cost function $c : \mathcal{K} \times \mathcal{K} \to \mathbb{R}$, the *optimal transport* distance [15] measures the cheapest way to transport the mass in probability measure μ_1 to match that in μ_2:

$$W_c(\mu_1, \mu_2) = \inf_{\gamma \in \Pi(\mu_1, \mu_2)} \int_{\mathcal{K}\times\mathcal{K}} c(\kappa_1, \kappa_2)\gamma(d\kappa_1, d\kappa_2) \tag{2}$$

where $\Pi(\mu_1, \mu_2)$ is the set of joint probability measures on $\mathcal{K}\times\mathcal{K}$ having μ_1 and μ_2 as marginals. An important case is that in which the cost is given by a metric $d_{\mathcal{K}}(\cdot, \cdot)$ or its p-th power $d^p_{\mathcal{K}}(\cdot, \cdot)$ with $p \geq 1$. In this case, (2) is called a *Wasserstein distance* [16], also known as the *earth mover's distance* [10]. In this paper, we only work with discrete measures. In the case of probability measures, these are histograms in the simplex $\Delta^{\mathcal{K}}$. When the ground truth y and the output of h both lie in the simplex $\Delta^{\mathcal{K}}$, we can define a Wasserstein loss.

Definition 3.1 (Exact Wasserstein Loss)**.** *For any $h_\theta \in \mathcal{H}$, $h_\theta : \mathcal{X} \to \Delta^{\mathcal{K}}$, let $h_\theta(\kappa|x) = h_\theta(x)_\kappa$ be the predicted value at element $\kappa \in \mathcal{K}$, given input $x \in \mathcal{X}$. Let $y(\kappa)$ be the ground truth value for κ given by the corresponding label y. Then we define the* exact Wasserstein loss *as*

$$W_p^p(h(\cdot|x), y(\cdot)) = \inf_{T \in \Pi(h(x), y)} \langle T, M \rangle \tag{3}$$

where $M \in \mathbb{R}_+^{K\times K}$ is the distance matrix $M_{\kappa,\kappa'} = d^p_{\mathcal{K}}(\kappa, \kappa')$, and the set of valid transport plans is

$$\Pi(h(x), y) = \{T \in \mathbb{R}_+^{K\times K} : T\mathbf{1} = h(x),\ T^\top \mathbf{1} = y\} \tag{4}$$

where $\mathbf{1}$ is the all-one vector.

W_p^p is the cost of the optimal plan for transporting the predicted mass distribution $h(x)$ to match the target distribution y. The penalty increases as more mass is transported over longer distances, according to the ground metric M.

Algorithm 1 Gradient of the Wasserstein loss

Given $h(x)$, y, λ, $\mathbf{K}$. (γ_a, γ_b if $h(x)$, y unnormalized.)
$u \leftarrow \mathbf{1}$
while u has not converged **do**

$$u \leftarrow \begin{cases} h(x) \oslash \left(\mathbf{K}\left(y \oslash \mathbf{K}^\top u\right)\right) & \text{if } h(x), y \text{ normalized} \\ h(x)^{\frac{\gamma_a\lambda}{\gamma_a\lambda+1}} \oslash \left(\mathbf{K}\left(y \oslash \mathbf{K}^\top u\right)^{\frac{\gamma_b\lambda}{\gamma_b\lambda+1}}\right)^{\frac{\gamma_a\lambda}{\gamma_a\lambda+1}} & \text{if } h(x), y \text{ unnormalized} \end{cases}$$

end while
If $h(x)$, y unnormalized: $v \leftarrow y^{\frac{\gamma_b\lambda}{\gamma_b\lambda+1}} \oslash \left(\mathbf{K}^\top u\right)^{\frac{\gamma_b\lambda}{\gamma_b\lambda+1}}$

$$\partial W_p^p / \partial h(x) \leftarrow \begin{cases} \frac{\log u}{\lambda} - \frac{\log u^\top \mathbf{1}}{\lambda K}\mathbf{1} & \text{if } h(x), y \text{ normalized} \\ \gamma_a\left(\mathbf{1} - (\mathsf{diag}(u)\mathbf{K}v) \oslash h(x)\right) & \text{if } h(x), y \text{ unnormalized} \end{cases}$$

4 Efficient optimization via entropic regularization

To do learning, we optimize the empirical risk minimization functional (1) by gradient descent. Doing so requires evaluating a descent direction for the loss, with respect to the predictions $h(x)$. Unfortunately, computing a subgradient of the exact Wasserstein loss (3), is quite costly, as follows.

The exact Wasserstein loss (3) is a linear program and a subgradient of its solution can be computed using Lagrange duality. The dual LP of (3) is

$$^dW_p^p(h(x), y) = \sup_{\alpha,\beta \in C_M} \alpha^\top h(x) + \beta^\top y, \quad C_M = \{(\alpha, \beta) \in \mathbb{R}^{K\times K} : \alpha_\kappa + \beta_{\kappa'} \leq M_{\kappa,\kappa'}\}. \quad (5)$$

As (3) is a linear program, at an optimum the values of the dual and the primal are equal (see, e.g. [17]), hence the dual optimal α is a subgradient of the loss with respect to its first argument.

Computing α is costly, as it entails solving a linear program with $O(K^2)$ contraints, with K being the dimension of the output space. This cost can be prohibitive when optimizing by gradient descent.

4.1 Entropic regularization of optimal transport

Cuturi [18] proposes a smoothed transport objective that enables efficient approximation of both the transport matrix in (3) and the subgradient of the loss. [18] introduces an entropic regularization term that results in a strictly convex problem:

$$^\lambda W_p^p(h(\cdot|x), y(\cdot)) = \inf_{T \in \Pi(h(x),y)} \langle T, M\rangle - \frac{1}{\lambda}H(T), \quad H(T) = -\sum_{\kappa,\kappa'} T_{\kappa,\kappa'} \log T_{\kappa,\kappa'}. \quad (6)$$

Importantly, the transport matrix that solves (6) is a *diagonal scaling* of a matrix $\mathbf{K} = e^{-\lambda M - 1}$:

$$T^* = \mathsf{diag}(u)\mathbf{K}\mathsf{diag}(v) \quad (7)$$

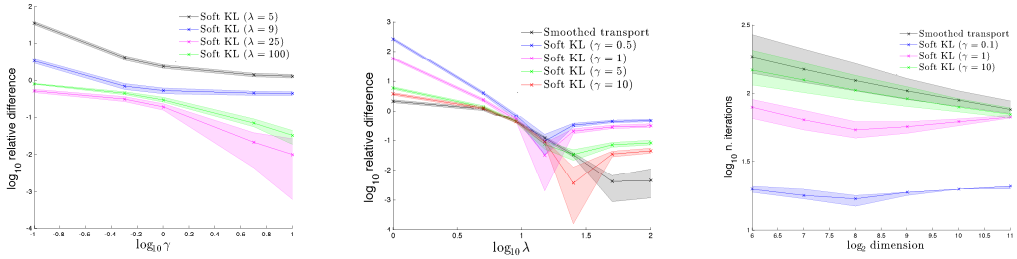
for $u = e^{\lambda\alpha}$ and $v = e^{\lambda\beta}$, where α and β are the Lagrange dual variables for (6).

Identifying such a matrix subject to equality constraints on the row and column sums is exactly a *matrix balancing* problem, which is well-studied in numerical linear algebra and for which efficient iterative algorithms exist [19]. [18] and [3] use the well-known Sinkhorn-Knopp algorithm.

4.2 Extending smoothed transport to the learning setting

When the output vectors $h(x)$ and y lie in the simplex, (6) can be used directly in place of (3), as (6) can approximate the exact Wasserstein distance closely for large enough λ [18]. In this case, the gradient α of the objective can be obtained from the optimal scaling vector u as $\alpha = \frac{\log u}{\lambda} - \frac{\log u^\top \mathbf{1}}{\lambda K}\mathbf{1}$. [1] A Sinkhorn iteration for the gradient is given in Algorithm 1.

[1] Note that α is only defined up to a constant shift: any upscaling of the vector u can be paired with a corresponding downscaling of the vector v (and vice versa) without altering the matrix T^*. The choice $\alpha = \frac{\log u}{\lambda} - \frac{\log u^\top \mathbf{1}}{\lambda K}\mathbf{1}$ ensures that α is tangent to the simplex.



(a) Convergence to smoothed transport.

(b) Approximation of exact Wasserstein.

(c) Convergence of alternating projections ($\lambda = 50$).

Figure 3: The relaxed transport problem (8) for unnormalized measures.

For many learning problems, however, a normalized output assumption is unnatural. In image segmentation, for example, the target shape is not naturally represented as a histogram. And even when the prediction and the ground truth are constrained to the simplex, the observed label can be subject to noise that violates the constraint.

There is more than one way to generalize optimal transport to unnormalized measures, and this is a subject of active study [20]. We will develop here a novel objective that deals effectively with the difference in total mass between $h(x)$ and y while still being efficient to optimize.

4.3 Relaxed transport

We propose a novel relaxation that extends smoothed transport to unnormalized measures. By replacing the equality constraints on the transport marginals in (6) with soft penalties with respect to KL divergence, we get an unconstrained approximate transport problem. The resulting objective is:

$$^{\lambda,\gamma_a,\gamma_b}W_{KL}(h(\cdot|x), y(\cdot)) = \min_{T\in\mathbb{R}_+^{K\times K}} \langle T, M\rangle - \frac{1}{\lambda}H(T) + \gamma_a \widetilde{\mathrm{KL}}\left(T\mathbf{1}\|h(x)\right) + \gamma_b \widetilde{\mathrm{KL}}\left(T^\top\mathbf{1}\|y\right) \quad (8)$$

where $\widetilde{\mathrm{KL}}(w\|z) = w^\top \log(w \oslash z) - \mathbf{1}^\top w + \mathbf{1}^\top z$ is the *generalized KL divergence* between $w, z \in \mathbb{R}_+^K$. Here $\oslash$ represents element-wise division. As with the previous formulation, the optimal transport matrix with respect to (8) is a diagonal scaling of the matrix $\mathbf{K}$.

Proposition 4.1. *The transport matrix T^* optimizing* (8) *satisfies $T^* = \mathsf{diag}(u)\mathbf{K}\mathsf{diag}(v)$, where $u = \left(h(x) \oslash T^*\mathbf{1}\right)^{\gamma_a\lambda}$, $v = \left(y \oslash (T^*)^\top\mathbf{1}\right)^{\gamma_b\lambda}$, and $\mathbf{K} = e^{-\lambda M - 1}$.*

And the optimal transport matrix is a fixed point for a Sinkhorn-like iteration. [2]

Proposition 4.2. *$T^* = \mathsf{diag}(u)\mathbf{K}\mathsf{diag}(v)$ optimizing* (8) *satisfies: i) $u = h(x)^{\frac{\gamma_a\lambda}{\gamma_a\lambda+1}} \odot (\mathbf{K}v)^{-\frac{\gamma_a\lambda}{\gamma_a\lambda+1}}$, and ii) $v = y^{\frac{\gamma_b\lambda}{\gamma_b\lambda+1}} \odot \left(\mathbf{K}^\top u\right)^{-\frac{\gamma_b\lambda}{\gamma_b\lambda+1}}$, where $\odot$ represents element-wise multiplication.*

Unlike the previous formulation, (8) is unconstrained with respect to $h(x)$. The gradient is given by $\nabla_{h(x)} W_{KL}(h(\cdot|x), y(\cdot)) = \gamma_a\left(\mathbf{1} - T^*\mathbf{1} \oslash h(x)\right)$. The iteration is given in Algorithm 1.

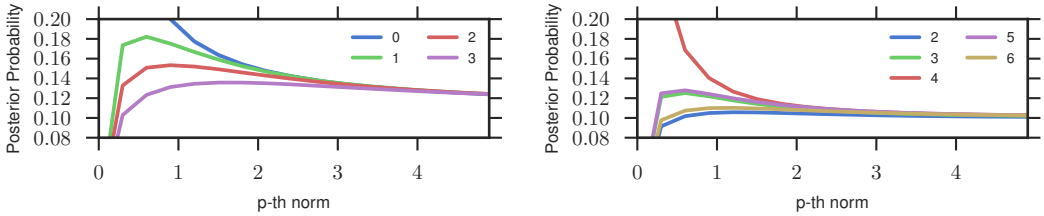
When restricted to normalized measures, the relaxed problem (8) approximates smoothed transport (6). Figure 3a shows, for normalized $h(x)$ and y, the relative distance between the values of (8) and (6) [3]. For λ large enough, (8) converges to (6) as γ_a and γ_b increase.

(8) also retains two properties of smoothed transport (6). Figure 3b shows that, for normalized outputs, the relaxed loss converges to the unregularized Wasserstein distance as λ, γ_a and γ_b increase [4]. And Figure 3c shows that convergence of the iterations in (4.2) is nearly independent of the dimension K of the output space.

[2] Note that, although the iteration suggested by Proposition 4.2 is observed empirically to converge (see Figure 3c, for example), we have not proven a guarantee that it will do so.

[3] In figures 3a-c, $h(x)$, y and M are generated as described in [18] section 5. In 3a-b, $h(x)$ and y have dimension 256. In 3c, convergence is defined as in [18]. Shaded regions are 95% intervals.

[4] The unregularized Wasserstein distance was computed using `FastEMD` [21].



(a) Posterior predictions for images of digit 0.

(b) Posterior predictions for images of digit 4.

Figure 4: MNIST example. Each curve shows the predicted probability for one digit, for models trained with different p values for the ground metric.

5 Statistical Properties of the Wasserstein loss

Let $S = ((x_1, y_1), \ldots, (x_N, y_N))$ be i.i.d. samples and $h_{\hat{\theta}}$ be the empirical risk minimizer

$$h_{\hat{\theta}} = \operatorname*{argmin}_{h_\theta \in \mathcal{H}} \left\{ \hat{\mathbb{E}}_S \left[W_p^p(h_\theta(\cdot|x), y) \right] = \frac{1}{N} \sum_{i=1}^{N} W_p^p(h_x\theta(\cdot|x_i), y_i) \right\}.$$

Further assume $\mathcal{H} = \mathfrak{s} \circ \mathcal{H}^o$ is the composition of a softmax $\mathfrak{s}$ and a base hypothesis space $\mathcal{H}^o$ of functions mapping into $\mathbb{R}^K$. The softmax layer outputs a prediction that lies in the simplex $\Delta^{\mathcal{K}}$.

Theorem 5.1. *For $p = 1$, and any $\delta > 0$, with probability at least $1 - \delta$, it holds that*

$$\mathbb{E}\left[W_1^1(h_{\hat{\theta}}(\cdot|x), y)\right] \leq \inf_{h_\theta \in \mathcal{H}} \mathbb{E}\left[W_1^1(h_\theta(\cdot|x), y)\right] + 32KC_M\mathfrak{R}_N(\mathcal{H}^o) + 2C_M\sqrt{\frac{\log(1/\delta)}{2N}} \quad (9)$$

with the constant $C_M = \max_{\kappa,\kappa'} M_{\kappa,\kappa'}$. $\mathfrak{R}_N(\mathcal{H}^o)$ is the Rademacher complexity *[22] measuring the complexity of the hypothesis space $\mathcal{H}^o$.*

The Rademacher complexity $\mathfrak{R}_N(\mathcal{H}^o)$ for commonly used models like neural networks and kernel machines [22] decays with the training set size. This theorem guarantees that the expected Wasserstein loss of the empirical risk minimizer approaches the best achievable loss for $\mathcal{H}$.

As an important special case, minimizing the empirical risk with Wasserstein loss is also good for multiclass classification. Let $y = \mathfrak{e}_\kappa$ be the "one-hot" encoded label vector for the groundtruth class.

Proposition 5.2. *In the multiclass classification setting, for $p = 1$ and any $\delta > 0$, with probability at least $1 - \delta$, it holds that*

$$\mathbb{E}_{x,\kappa}\left[d_{\mathcal{K}}(\kappa_{\hat{\theta}}(x), \kappa)\right] \leq \inf_{h_\theta \in \mathcal{H}} K\mathbb{E}[W_1^1(h_\theta(x), y)] + 32K^2C_M\mathfrak{R}_N(\mathcal{H}^o) + 2C_MK\sqrt{\frac{\log(1/\delta)}{2N}} \quad (10)$$

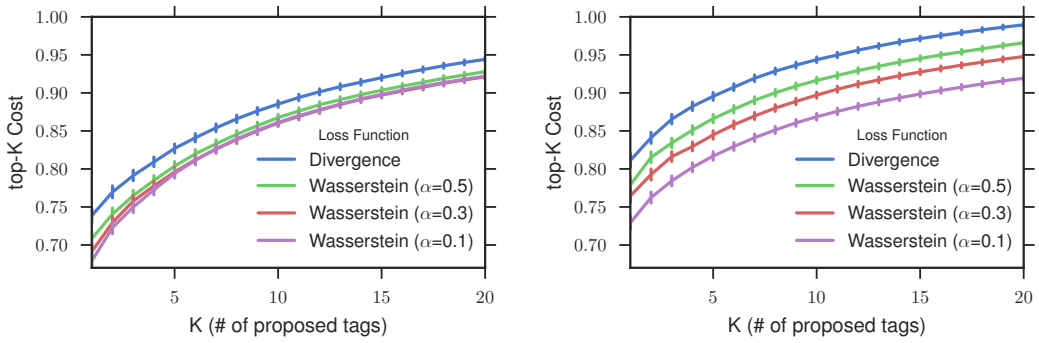
where the predictor is $\kappa_{\hat{\theta}}(x) = \operatorname{argmax}_\kappa h_{\hat{\theta}}(\kappa|x)$, with $h_{\hat{\theta}}$ being the empirical risk minimizer.

Note that instead of the classification error $\mathbb{E}_{x,\kappa}[\mathbb{1}\{\kappa_{\hat{\theta}}(x) \neq \kappa\}]$, we actually get a bound on the expected semantic distance between the prediction and the groundtruth.

6 Empirical study

6.1 Impact of the ground metric

In this section, we show that the Wasserstein loss encourages smoothness with respect to an artificial metric on the MNIST handwritten digit dataset. This is a multi-class classification problem with output dimensions corresponding to the 10 digits, and we apply a ground metric $d_p(\kappa, \kappa') = |\kappa - \kappa'|^p$, where $\kappa, \kappa' \in \{0, \ldots, 9\}$ and $p \in [0, \infty)$. This metric encourages the recognized digit to be *numerically* close to the true one. We train a model independently for each value of p and plot the average predicted probabilities of the different digits on the test set in Figure 4.



(a) Original Flickr tags dataset.

(b) Reduced-redundancy Flickr tags dataset.

Figure 5: Top-K cost comparison of the proposed loss (Wasserstein) and the baseline (Divergence).

Note that as $p \to 0$, the metric approaches the $0-1$ metric $d_0(\kappa, \kappa') = \mathbb{1}_{\kappa \neq \kappa'}$, which treats all incorrect digits as being equally unfavorable. In this case, as can be seen in the figure, the predicted probability of the true digit goes to 1 while the probability for all other digits goes to 0. As p increases, the predictions become more evenly distributed over the neighboring digits, converging to a uniform distribution as $p \to \infty$ [5].

6.2 Flickr tag prediction

We apply the Wasserstein loss to a real world multi-label learning problem, using the recently released Yahoo/Flickr Creative Commons 100M dataset [23]. [6] Our goal is *tag prediction*: we select 1000 descriptive tags along with two random sets of 10,000 images each, associated with these tags, for training and testing. We derive a distance metric between tags by using `word2vec` [24] to embed the tags as unit vectors, then taking their Euclidean distances. To extract image features we use `MatConvNet` [25]. Note that the set of tags is highly redundant and often many semantically equivalent or similar tags can apply to an image. The images are also partially tagged, as different users may prefer different tags. We therefore measure the prediction performance by the *top-K cost*, defined as $C_K = 1/K \sum_{k=1}^{K} \min_j d_{\mathcal{K}}(\hat{\kappa}_k, \kappa_j)$, where $\{\kappa_j\}$ is the set of groundtruth tags, and $\{\hat{\kappa}_k\}$ are the tags with highest predicted probability. The standard AUC measure is also reported.

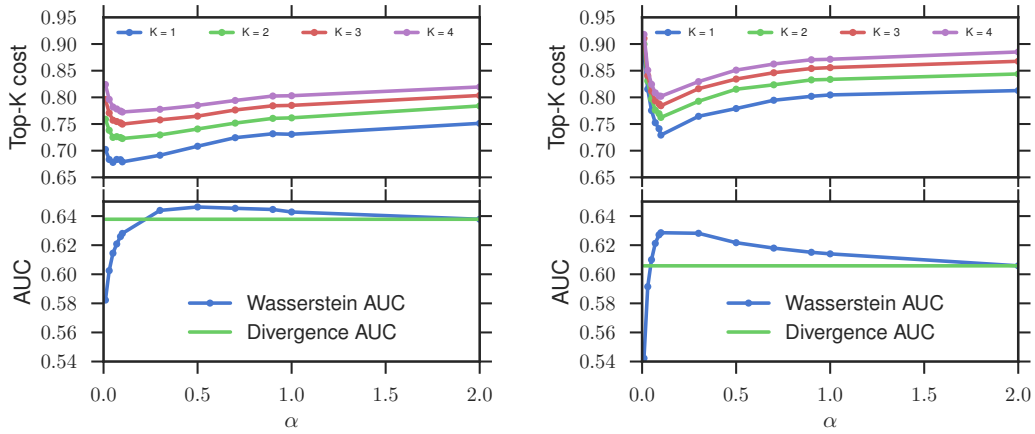
We find that a linear combination of the Wasserstein loss W_p^p and the standard multiclass logistic loss KL yields the best prediction results. Specifically, we train a linear model by minimizing $W_p^p + \alpha \mathsf{KL}$ on the training set, where α controls the relative weight of KL. Note that KL taken alone is our baseline in these experiments. Figure 5a shows the top-K cost on the test set for the combined loss and the baseline KL loss. We additionally create a second dataset by removing redundant labels from the original dataset: this simulates the potentially more difficult case in which a single user tags each image, by selecting one tag to apply from amongst each cluster of applicable, semantically similar tags. Figure 3b shows that performance for both algorithms decreases on the harder dataset, while the combined Wasserstein loss continues to outperform the baseline.

In Figure 6, we show the effect on performance of varying the weight α on the KL loss. We observe that the optimum of the top-K cost is achieved when the Wasserstein loss is weighted more heavily than at the optimum of the AUC. This is consistent with a semantic smoothing effect of Wasserstein, which during training will favor mispredictions that are semantically similar to the ground truth, sometimes at the cost of lower AUC [7]. We finally show two selected images from the test set in Figure 7. These illustrate cases in which both algorithms make predictions that are semantically relevant, despite overlapping very little with the ground truth. The image on the left shows errors made by both algorithms. More examples can be found in the appendix.

[5] To avoid numerical issues, we scale down the ground metric such that all of the distance values are in the interval $[0, 1)$.

[6] The dataset used here is available at `http://cbcl.mit.edu/wasserstein`.

[7] The Wasserstein loss can achieve a similar trade-off by choosing the metric parameter p, as discussed in Section 6.1. However, the relationship between p and the smoothing behavior is complex and it can be simpler to implement the trade-off by combining with the KL loss.



(a) Original Flickr tags dataset.

(b) Reduced-redundancy Flickr tags dataset.

Figure 6: Trade-off between semantic smoothness and maximum likelihood.

(a) **Flickr user tags**: street, parade, dragon; **our proposals**: people, protest, parade; **baseline proposals**: music, car, band.

(b) **Flickr user tags**: water, boat, reflection, sunshine; **our proposals**: water, river, lake, summer; **baseline proposals**: river, water, club, nature.

Figure 7: Examples of images in the Flickr dataset. We show the groundtruth tags and as well as tags proposed by our algorithm and the baseline.

7 Conclusions and future work

In this paper we have described a loss function for learning to predict a non-negative measure over a finite set, based on the Wasserstein distance. Although optimizing with respect to the exact Wasserstein loss is computationally costly, an approximation based on entropic regularization is efficiently computed. We described a learning algorithm based on this regularization and we proposed a novel extension of the regularized loss to unnormalized measures that preserves its efficiency. We also described a statistical learning bound for the loss. The Wasserstein loss can encourage smoothness of the predictions with respect to a chosen metric on the output space, and we demonstrated this property on a real-data tag prediction problem, showing improved performance over a baseline that doesn't incorporate the metric.

An interesting direction for future work may be to explore the connection between the Wasserstein loss and Markov random fields, as the latter are often used to encourage smoothness of predictions, via inference at prediction time.

References

[1] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *CVPR (to appear)*, 2015.

[2] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 2015.

[3] Marco Cuturi and Arnaud Doucet. Fast Computation of Wasserstein Barycenters. *ICML*, 2014.

[4] Justin Solomon, Raif M Rustamov, Leonidas J Guibas, and Adrian Butscher. Wasserstein Propagation for Semi-Supervised Learning. In *ICML*, pages 306–314, 2014.

[5] Michael H Coen, M Hidayath Ansari, and Nathanael Fillmore. Comparing Clusterings in Space. *ICML*, pages 231–238, 2010.

[6] Lorenzo Rosasco Mauricio A. Alvarez and Neil D. Lawrence. Kernels for vector-valued functions: A review. *Foundations and Trends in Machine Learning*, 4(3):195–266, 2011.

[7] Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.

[8] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *ICLR*, 2015.

[9] Marco Cuturi, Gabriel Peyré, and Antoine Rolet. A Smoothed Dual Approach for Variational Wasserstein Problems. *arXiv.org*, March 2015.

[10] Yossi Rubner, Carlo Tomasi, and Leonidas J Guibas. The earth mover's distance as a metric for image retrieval. *IJCV*, 40(2):99–121, 2000.

[11] Kristen Grauman and Trevor Darrell. Fast contour matching using approximate earth mover's distance. In *CVPR*, 2004.

[12] S Shirdhonkar and D W Jacobs. Approximate earth mover's distance in linear time. In *CVPR*, 2008.

[13] Herbert Edelsbrunner and Dmitriy Morozov. Persistent homology: Theory and practice. In *Proceedings of the European Congress of Mathematics*, 2012.

[14] Federico Bassetti, Antonella Bodini, and Eugenio Regazzini. On minimum kantorovich distance estimators. *Stat. Probab. Lett.*, 76(12):1298–1302, 1 July 2006.

[15] Cédric Villani. *Optimal Transport: Old and New*. Springer Berlin Heidelberg, 2008.

[16] Vladimir I Bogachev and Aleksandr V Kolesnikov. The Monge-Kantorovich problem: achievements, connections, and perspectives. *Russian Math. Surveys*, 67(5):785, 10 2012.

[17] Dimitris Bertsimas, John N. Tsitsiklis, and John Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, Boston, third printing edition, 1997.

[18] Marco Cuturi. Sinkhorn Distances: Lightspeed Computation of Optimal Transport. *NIPS*, 2013.

[19] Philip A Knight and Daniel Ruiz. A fast algorithm for matrix balancing. *IMA Journal of Numerical Analysis*, 33(3):drs019–1047, October 2012.

[20] Lenaic Chizat, Gabriel Peyré, Bernhard Schmitzer, and François-Xavier Vialard. Unbalanced Optimal Transport: Geometry and Kantorovich Formulation. *arXiv.org*, August 2015.

[21] Ofir Pele and Michael Werman. Fast and robust Earth Mover's Distances. *ICCV*, pages 460–467, 2009.

[22] Peter L Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *JMLR*, 3:463–482, March 2003.

[23] Bart Thomee, David A. Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. The new data and new challenges in multimedia research. *arXiv preprint arXiv:1503.01817*, 2015.

[24] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013.

[25] A. Vedaldi and K. Lenc. MatConvNet – Convolutional Neural Networks for MATLAB. *CoRR*, abs/1412.4564, 2014.

[26] M. Ledoux and M. Talagrand. *Probability in Banach Spaces: Isoperimetry and Processes*. Classics in Mathematics. Springer Berlin Heidelberg, 2011.

[27] Clark R. Givens and Rae Michael Shortt. A class of wasserstein metrics for probability distributions. *Michigan Math. J.*, 31(2):231–240, 1984.

A Relaxed transport

Equation (8) gives the relaxed transport objective as

$$^{\lambda,\gamma_a,\gamma_b}W_{KL}(h(\cdot|x),y(\cdot)) = \min_{T\in\mathbb{R}_+^{K\times K}} \langle T, M\rangle - \frac{1}{\lambda}H(T) + \gamma_a \widetilde{\mathrm{KL}}\left(T\mathbf{1}\|h(x)\right) + \gamma_b \widetilde{\mathrm{KL}}\left(T^\top\mathbf{1}\|y\right)$$

with $\widetilde{\mathrm{KL}}\left(w\|z\right) = w^\top \log(w \oslash z) - \mathbf{1}^\top w + \mathbf{1}^\top z$.

Proof of Proposition 4.1. The first order condition for T^* optimizing (8) is

$$\begin{aligned}
&M_{ij} + \frac{1}{\lambda}\left(\log T^*_{ij} + 1\right) + \gamma_a \left(\log T^*\mathbf{1} \oslash h(x)\right)_i + \gamma_b \left(\log (T^*)^\top\mathbf{1} \oslash y\right)_j = 0.\\
\Rightarrow &\log T^*_{ij} + \gamma_a\lambda \log\left(T^*\mathbf{1} \oslash h(x_i)\right)_i + \gamma_b\lambda \log\left((T^*)^\top\mathbf{1} \oslash y_j\right)_j = -\lambda M_{ij} - 1\\
\Rightarrow &T^*_{ij}\left(T^*\mathbf{1} \oslash h(x)\right)_i^{\gamma_a\lambda}\left((T^*)^\top\mathbf{1} \oslash y\right)_j^{\gamma_b\lambda} = \exp\left(-\lambda M_{ij} - 1\right)\\
\Rightarrow &T^*_{ij} = \left(h(x) \oslash T^*\mathbf{1}\right)_i^{\gamma_a\lambda}\left(y \oslash (T^*)^\top\mathbf{1}\right)_j^{\gamma_b\lambda}\exp\left(-\lambda M_{ij} - 1\right)
\end{aligned}$$

Hence T^* (if it exists) is a diagonal scaling of $\mathbf{K} = \exp\left(-\lambda M - 1\right)$.

□

Proof of Proposition 4.2. Let $u = \left(h(x) \oslash T^*\mathbf{1}\right)^{\gamma_a\lambda}$ and $v = \left(y \oslash (T^*)^\top\mathbf{1}\right)^{\gamma_b\lambda}$, so $T^* = \mathsf{diag}(u)\mathbf{K}\mathsf{diag}(v)$. We have

$$\begin{aligned}
&T^*\mathbf{1} = \mathsf{diag}(u)\mathbf{K}v\\
\Rightarrow &(T^*\mathbf{1})^{\gamma_a\lambda+1} = h(x)^{\gamma_a\lambda} \odot \mathbf{K}v
\end{aligned}$$

where we substituted the expression for u. Re-writing $T^*\mathbf{1}$,

$$\begin{aligned}
&(\mathsf{diag}(u)\mathbf{K}v)^{\gamma_a\lambda+1} = \mathsf{diag}(h(x)^{\gamma_a\lambda})\mathbf{K}v\\
\Rightarrow &u^{\gamma_a\lambda+1} = h(x)^{\gamma_a\lambda} \odot (\mathbf{K}v)^{-\gamma_a\lambda}\\
\Rightarrow &u = h(x)^{\frac{\gamma_a\lambda}{\gamma_a\lambda+1}} \odot (\mathbf{K}v)^{-\frac{\gamma_a\lambda}{\gamma_a\lambda+1}}.
\end{aligned}$$

A symmetric argument shows that $v = y^{\frac{\gamma_b\lambda}{\gamma_b\lambda+1}} \odot (\mathbf{K}^\top u)^{-\frac{\gamma_b\lambda}{\gamma_b\lambda+1}}$. □

B Statistical Learning Bounds

We establish the proof of Theorem 5.1 in this section. For simpler notation, for a sequence $S = ((x_1, y_1), \ldots, (x_N, y_N))$ of i.i.d. training samples, we denote the empirical risk $\hat{R}_S$ and risk R as

$$\hat{R}_S(h_\theta) = \hat{\mathbb{E}}_S\left[W_p^p(h_\theta(\cdot|x), y(\cdot))\right], \quad R(h_\theta) = \mathbb{E}\left[W_p^p(h_\theta(\cdot|x), y(\cdot))\right] \tag{11}$$

Lemma B.1. *Let $h_{\hat{\theta}}, h_{\theta^*} \in \mathcal{H}$ be the minimizer of the empirical risk $\hat{R}_S$ and expected risk R, respectively. Then*

$$R(h_{\hat{\theta}}) \le R(h_{\theta^*}) + 2\sup_{h\in\mathcal{H}} |R(h) - \hat{R}_S(h)|$$

Proof. By the optimality of $h_{\hat{\theta}}$ for $\hat{R}_S$,

$$\begin{aligned}
R(h_{\hat{\theta}}) - R(h_{\theta^*}) &= R(h_{\hat{\theta}}) - \hat{R}_S(h_{\hat{\theta}}) + \hat{R}_S(h_{\hat{\theta}}) - R(h_{\theta^*})\\
&\le R(h_{\hat{\theta}}) - \hat{R}_S(h_{\hat{\theta}}) + \hat{R}_S(h_{\theta^*}) - R(h_{\theta^*})\\
&\le 2\sup_{h\in\mathcal{H}} |R(h) - \hat{R}_S(h)|
\end{aligned}$$

□

Therefore, to bound the risk for $h_{\hat{\theta}}$, we need to establish uniform concentration bounds for the Wasserstein loss. Towards that goal, we define a space of loss functions induced by the hypothesis space $\mathcal{H}$ as

$$\mathcal{L} = \left\{ \ell_\theta : (x, y) \mapsto W_p^p(h_\theta(\cdot|x), y(\cdot)) : h_\theta \in \mathcal{H} \right\} \tag{12}$$

The uniform concentration will depends on the "complexity" of $\mathcal{L}$, which is measured by the empirical *Rademacher complexity* defined below.

Definition B.2 (Rademacher Complexity [22]). *Let $\mathcal{G}$ be a family of mapping from $\mathcal{Z}$ to $\mathbb{R}$, and $S = (z_1, \ldots, z_N)$ a fixed sample from $\mathcal{Z}$. The* empirical Rademacher complexity *of $\mathcal{G}$ with respect to S is defined as*

$$\hat{\mathfrak{R}}_S(\mathcal{G}) = \mathbb{E}_\sigma \left[\sup_{g \in \mathcal{G}} \frac{1}{N} \sum_{i=1}^{n} \sigma_i g(z_i) \right] \tag{13}$$

where $\sigma = (\sigma_1, \ldots, \sigma_N)$, with σ_i's independent uniform random variables taking values in $\{+1, -1\}$. σ_i's are called the Rademacher random variables. The Rademacher complexity *is defined by taking expectation with respect to the samples S,*

$$\mathfrak{R}_N(\mathcal{G}) = \mathbb{E}_S \left[\hat{\mathfrak{R}}_S(\mathcal{G}) \right] \tag{14}$$

Theorem B.3. *For any $\delta > 0$, with probability at least $1 - \delta$, the following holds for all $\ell_\theta \in \mathcal{L}$,*

$$\mathbb{E}[\ell_\theta] - \hat{\mathbb{E}}_S[\ell_\theta] \leq 2\mathfrak{R}_N(\mathcal{L}) + \sqrt{\frac{C_M^2 \log(1/\delta)}{2N}} \tag{15}$$

with the constant $C_M = \max_{\kappa,\kappa'} M_{\kappa,\kappa'}$.

By the definition of $\mathcal{L}$, $\mathbb{E}[\ell_\theta] = R(h_\theta)$ and $\hat{\mathbb{E}}_S[\ell_\theta] = \hat{R}_S[h_\theta]$. Therefore, this theorem provides a uniform control for the deviation of the empirical risk from the risk.

Theorem B.4 (McDiarmid's Inequality). *Let $S = \{X_1, \ldots, X_N\} \subset \mathscr{X}$ be N i.i.d. random variables. Assume there exists $C > 0$ such that $f : \mathscr{X}^N \to \mathbb{R}$ satisfies the following stability condition*

$$|f(x_1, \ldots, x_i, \ldots, x_N) - f(x_1, \ldots, x_i', \ldots, x_N)| \leq C \tag{16}$$

for all $i = 1, \ldots, N$ and any $x_1, \ldots, x_N, x_i' \in \mathscr{X}$. Then for any $\varepsilon > 0$, denoting $f(X_1, \ldots, X_N)$ by $f(S)$, it holds that

$$\mathbb{P}\left(f(S) - \mathbb{E}[f(S)] \geq \varepsilon \right) \leq \exp\left(-\frac{2\varepsilon^2}{NC^2} \right) \tag{17}$$

Lemma B.5. *Let the constant $C_M = \max_{\kappa,\kappa'} M_{\kappa,\kappa'}$, then $0 \leq W_p^p(\cdot, \cdot) \leq C_M$.*

Proof. For any $h(\cdot|x)$ and $y(\cdot)$, let $T^* \in \Pi(h(x), y)$ be the optimal transport plan that solves (3), then

$$W_p^p(h(x), y) = \langle T^*, M \rangle \leq C_M \sum_{\kappa,\kappa'} T_{\kappa,\kappa'} = C_M$$

□

Proof of Theorem B.3. For any $\ell_\theta \in \mathcal{L}$, note the empirical expectation is the empirical risk of the corresponding h_θ:

$$\hat{E}_S[\ell_\theta] = \frac{1}{N} \sum_{i=1}^{N} \ell_\theta(x_i, y_i) = \frac{1}{N} \sum_{i=1}^{N} W_p^p(h_\theta(\cdot|x_i), y_i(\cdot)) = \hat{R}_S(h_\theta)$$

Similarly, $\mathbb{E}[\ell_\theta] = R(h_\theta)$. Let

$$\Phi(S) = \sup_{\ell \in \mathcal{L}} \mathbb{E}[\ell] - \hat{\mathbb{E}}_S[\ell] \tag{18}$$

Let S' be S with the i-th sample replaced by (x_i', y_i'), by Lemma B.5, it holds that

$$\Phi(S) - \Phi(S') \leq \sup_{\ell \in \mathcal{L}} \hat{\mathbb{E}}_{S'}[\ell] - \hat{\mathbb{E}}_S[\ell] = \sup_{h_\theta \in \mathcal{H}} \frac{W_p^p(h_\theta(x_i'), y_i') - W_p^p(h_\theta(x_i), y_i)}{N} \leq \frac{C_M}{N}$$

Similarly, we can show $\Phi(S') - \Phi(S) \leq C_M/N$, thus $|\Phi(S') - \Phi(S)| \leq C_M/N$. By Theorem B.4, for any $\delta > 0$, with probability at least $1 - \delta$, it holds that

$$\Phi(S) \leq \mathbb{E}[\Phi(S)] + \sqrt{\frac{C_M^2 \log(1/\delta)}{2N}} \tag{19}$$

To bound $\mathbb{E}[\Phi(S)]$, by Jensen's inequality,

$$\mathbb{E}_S[\Phi(S)] = \mathbb{E}_S\left[\sup_{\ell \in \mathcal{L}} \mathbb{E}[\ell] - \hat{\mathbb{E}}_S[\ell]\right] = \mathbb{E}_S\left[\sup_{\ell \in \mathcal{L}} \mathbb{E}_{S'}\left[\hat{\mathbb{E}}_{S'}[\ell] - \hat{\mathbb{E}}_S[\ell]\right]\right] \leq \mathbb{E}_{S,S'}\left[\sup_{\ell \in \mathcal{L}} \hat{E}_{S'}[\ell] - \hat{E}_S[\ell]\right]$$

Here S' is another sequence of i.i.d. samples, usually called *ghost samples*, that is only used for analysis. Now we introduce the Rademacher variables σ_i, since the role of S and S' are completely symmetric, it follows

$$\begin{aligned}
\mathbb{E}_S[\Phi(S)] &\leq \mathbb{E}_{S,S',\sigma}\left[\sup_{\ell \in \mathcal{L}} \frac{1}{N}\sum_{i=1}^N \sigma_i(\ell(x_i', y_i') - \ell(x_i, y_i))\right] \\
&\leq \mathbb{E}_{S',\sigma}\left[\sup_{\ell \in \mathcal{L}} \frac{1}{N}\sum_{i=1}^N \sigma_i \ell(x_i', y_i')\right] + \mathbb{E}_{S,\sigma}\left[\sup_{\ell \in \mathcal{L}} \frac{1}{N}\sum_{i=1}^N -\sigma_i \ell(x_i, y_i)\right] \\
&= \mathbb{E}_S\left[\hat{\mathfrak{R}}_S(\mathcal{L})\right] + \mathbb{E}_{S'}\left[\hat{\mathfrak{R}}_{S'}(\mathcal{L})\right] \\
&= 2\mathfrak{R}_N(\mathcal{L})
\end{aligned}$$

The conclusion follows by combing (18) and (19). □

To finish the proof of Theorem 5.1, we combine Lemma B.1 and Theorem B.3, and relate $\mathfrak{R}_N(\mathcal{L})$ to $\mathfrak{R}_N(\mathcal{H})$ via the following generalized Talagrand's lemma [26].

Lemma B.6. *Let $\mathcal{F}$ be a class of real functions, and $\mathcal{H} \subset \mathcal{F} = \mathcal{F}_1 \times \ldots \times \mathcal{F}_K$ be a K-valued function class. If $\mathfrak{m} : \mathbb{R}^K \to \mathbb{R}$ is a $L_\mathfrak{m}$-Lipschitz function and $\mathfrak{m}(0) = 0$, then $\hat{\mathfrak{R}}_S(\mathfrak{m} \circ \mathcal{H}) \leq 2L_\mathfrak{m} \sum_{k=1}^K \hat{\mathfrak{R}}_S(\mathcal{F}_k)$.*

Theorem B.7 (Theorem 6.15 of [15])**.** *Let μ and ν be two probability measures on a Polish space $(\mathcal{K}, d_\mathcal{K})$. Let $p \in [1, \infty)$ and $\kappa_0 \in \mathcal{K}$. Then*

$$W_p(\mu, \nu) \leq 2^{1/p'}\left(\int_\mathcal{K} d_\mathcal{K}(\kappa_0, \kappa) d|\mu - \nu|(\kappa)\right)^{1/p}, \quad \frac{1}{p} + \frac{1}{p'} = 1 \tag{20}$$

Corollary B.8. *The Wasserstein loss is Lipschitz continuous in the sense that for any $h_\theta \in \mathcal{H}$, and any $(x, y) \in \mathcal{X} \times \mathcal{Y}$,*

$$W_p^p(h_\theta(\cdot|x), y) \leq 2^{p-1} C_M \sum_{\kappa \in \mathcal{K}} |h_\theta(\kappa|x) - y(\kappa)| \tag{21}$$

In particular, when $p = 1$, we have

$$W_1^1(h_\theta(\cdot|x), y) \leq C_M \sum_{\kappa \in \mathcal{K}} |h_\theta(\kappa|x) - y(\kappa)| \tag{22}$$

We cannot apply Lemma B.6 directly to the Wasserstein loss class, because the Wasserstein loss is only defined on probability distributions, so 0 is not a valid input. To get around this problem, we assume the hypothesis space $\mathcal{H}$ used in learning is of the form

$$\mathcal{H} = \{\mathfrak{s} \circ h^o : h^o \in \mathcal{H}^o\} \tag{23}$$

where $\mathcal{H}^o$ is a function class that maps into $\mathbb{R}^K$, and $\mathfrak{s}$ is the softmax function defined as $\mathfrak{s}(o) = (\mathfrak{s}_1(o), \ldots, \mathfrak{s}_K(o))$, with

$$\mathfrak{s}_k(o) = \frac{e^{o_k}}{\sum_j e^{o_j}}, \quad k = 1, \ldots, K \tag{24}$$

The softmax layer produce a valid probability distribution from arbitrary input, and this is consistent with commonly used models such as Logistic Regression and Neural Networks. By working with the $\log$ of the groundtruth labels, we can also add a softmax layer to the labels.

Lemma B.9 (Proposition 2 of [27]). *The Wasserstein distances $W_p(\cdot,\cdot)$ are metrics on the space of probability distributions of $\mathcal{K}$, for all $1 \le p \le \infty$.*

Proposition B.10. *The map $\iota : \mathbb{R}^K \times \mathbb{R}^K \to \mathbb{R}$ defined by $\iota(y, y') = W_1^1(\mathfrak{s}(y), \mathfrak{s}(y'))$ satisfies*

$$|\iota(y, y') - \iota(\bar{y}, \bar{y}')| \le 4C_M \|(y, y') - (\bar{y}, \bar{y}')\|_2 \tag{25}$$

for any $(y, y'), (\bar{y}, \bar{y}') \in \mathbb{R}^K \times \mathbb{R}^K$. And $\iota(0, 0) = 0$.

Proof. For any $(y, y'), (\bar{y}, \bar{y}') \in \mathbb{R}^K \times \mathbb{R}^K$, by Lemma B.9, we can use triangle inequality on the Wasserstein loss,

$$|\iota(y, y') - \iota(\bar{y}, \bar{y}')| = |\iota(y, y') - \iota(\bar{y}, y') + \iota(\bar{y}, y') - \iota(\bar{y}, \bar{y}')| \le \iota(y, \bar{y}) + \iota(y', \bar{y}')$$

Following Corollary B.8, it continues as

$$|\iota(y, y') - \iota(\bar{y}, \bar{y}')| \le C_M \left(\|\mathfrak{s}(y) - \mathfrak{s}(\bar{y})\|_1 + \|\mathfrak{s}(y') - \mathfrak{s}(\bar{y}')\|_1\right) \tag{26}$$

Note for each $k = 1, \ldots, K$, the gradient $\nabla_y \mathfrak{s}_k$ satisfies

$$\|\nabla_y \mathfrak{s}_k\|_2 = \left\| \left(\frac{\partial \mathfrak{s}_k}{\partial y_j} \right)_{j=1}^K \right\|_2 = \left\| (\delta_{kj}\mathfrak{s}_k - \mathfrak{s}_k \mathfrak{s}_j)_{j=1}^K \right\|_2 = \sqrt{\mathfrak{s}_k^2 \sum_{j=1}^K \mathfrak{s}_j^2 + \mathfrak{s}_k^2(1 - 2\mathfrak{s}_k)} \tag{27}$$

By mean value theorem, $\exists \alpha \in [0, 1]$, such that for $y_\theta = \alpha y + (1 - \alpha)\bar{y}$, it holds that

$$\|\mathfrak{s}(y) - \mathfrak{s}(\bar{y})\|_1 = \sum_{k=1}^K \left| \langle \nabla_y \mathfrak{s}_k|_{y=y_{\alpha_k}}, y - \bar{y} \rangle \right| \le \sum_{k=1}^K \|\nabla_y \mathfrak{s}_k|_{y=y_{\alpha_k}}\|_2 \|y - \bar{y}\|_2 \le 2\|y - \bar{y}\|_2$$

because by (27), and the fact that $\sqrt{\sum_j \mathfrak{s}_j^2} \le \sum_j \mathfrak{s}_j = 1$ and $\sqrt{a + b} \le \sqrt{a} + \sqrt{b}$ for $a, b \ge 0$, it holds

$$\begin{aligned} \sum_{k=1}^K \|\nabla_y \mathfrak{s}_k\|_2 &= \sum_{k:\mathfrak{s}_k \le 1/2} \|\nabla_y \mathfrak{s}_k\|_2 + \sum_{k:\mathfrak{s}_k > 1/2} \|\nabla_y \mathfrak{s}_k\|_2 \\ &\le \sum_{k:\mathfrak{s}_k \le 1/2} \left(\mathfrak{s}_k + \mathfrak{s}_k\sqrt{1 - 2\mathfrak{s}_k}\right) + \sum_{k:\mathfrak{s}_k > 1/2} \mathfrak{s}_k \le \sum_{k=1}^K 2\mathfrak{s}_k = 2 \end{aligned}$$

Similarly, we have $\|\mathfrak{s}(y') - \mathfrak{s}(\bar{y}')\|_1 \le 2\|y' - \bar{y}'\|_2$, so from (26), we know

$$|\iota(y, y') - \iota(\bar{y}, \bar{y}')| \le 2C_M(\|y - \bar{y}\|_2 + \|y' - \bar{y}'\|_2) \le 2\sqrt{2}C_M \left(\|y - \bar{y}\|_2^2 + \|y' - \bar{y}'\|_2^2\right)^{1/2}$$

then (25) follows immediately. The second conclusion follows trivially as $\mathfrak{s}$ maps the zero vector to a uniform distribution. □

Proof of Theorem 5.1. Consider the loss function space preceded with a softmax layer

$$\mathcal{L} = \{\iota_\theta : (x, y) \mapsto W_1^1(\mathfrak{s}(h_\theta^o(x)), \mathfrak{s}(y)) : h_\theta^o \in \mathcal{H}^o\}$$

We apply Lemma B.6 to the $4C_M$-Lipschitz continuous function ι in Proposition B.10 and the function space

$$\underbrace{\mathcal{H}^o \times \ldots \times \mathcal{H}^o}_{K \text{ copies}} \times \underbrace{\mathcal{I} \times \ldots \times \mathcal{I}}_{K \text{ copies}}$$

with $\mathcal{I}$ a singleton function space with only the identity map. It holds

$$\hat{\mathfrak{R}}_S(\mathcal{L}) \le 8C_M \left(K\hat{\mathfrak{R}}_S(\mathcal{H}^o) + K\hat{\mathfrak{R}}_S(\mathcal{I})\right) = 8KC_M\hat{\mathfrak{R}}_S(\mathcal{H}^o) \tag{28}$$

because for the identity map, and a sample $S = (y_1, \ldots, y_N)$, we can calculate

$$\hat{\mathfrak{R}}_S(\mathcal{I}) = \mathbb{E}_\sigma \left[\sup_{f \in \mathcal{I}} \frac{1}{N} \sum_{i=1}^N \sigma_i f(y_i) \right] = \mathbb{E}_\sigma \left[\frac{1}{N} \sum_{i=1}^N \sigma_i y_i \right] = 0$$

The conclusion of the theorem follows by combining (28) with Theorem B.3 and Lemma B.1. □

C Connection with multiclass classification

Proof of Proposition 5.2. Given that the label is a "one-hot" vector $y = \mathbb{e}_\kappa$, the set of transport plans (4) degenerates. Specifically, the constraint $T^\top \mathbf{1} = \mathbb{e}_\kappa$ means that only the κ-th column of T can be non-zero. Furthermore, the constraint $T\mathbf{1} = h_{\hat{\theta}}(\cdot|x)$ ensures that the κ-th column of T actually equals $h_{\hat{\theta}}(\cdot|x)$. In other words, the set $\Pi(h_{\hat{\theta}(\cdot|x)}, \mathbb{e}_\kappa)$ contains only one feasible transport plan, so (3) can be computed directly as

$$W_p^p(h_{\hat{\theta}}(\cdot|x), \mathbb{e}_\kappa) = \sum_{\kappa' \in \mathcal{K}} M_{\kappa',\kappa} h_{\hat{\theta}}(\kappa'|x) = \sum_{\kappa' \in \mathcal{K}} d_{\mathcal{K}}^p(\kappa', \kappa) h_{\hat{\theta}}(\kappa'|x)$$

Now let $\hat{\kappa} = \operatorname{argmax}_\kappa h_{\hat{\theta}}(\kappa|x)$ be the prediction, we have

$$h_{\hat{\theta}}(\hat{\kappa}|x) = 1 - \sum_{\kappa \neq \hat{\kappa}} h_{\hat{\theta}}(\kappa|x) \geq 1 - \sum_{\kappa \neq \hat{\kappa}} h_{\hat{\theta}}(\hat{\kappa}|x) = 1 - (K-1) h_{\hat{\theta}}(\hat{\kappa}|x)$$

Therefore, $h_{\hat{\theta}}(\hat{\kappa}|x) \geq 1/K$, so

$$W_p^p(h_{\hat{\theta}}(\cdot|x), \mathbb{e}_\kappa) \geq d_{\mathcal{K}}^p(\hat{\kappa}, \kappa) h_{\hat{\theta}}(\hat{\kappa}|x) \geq d_{\mathcal{K}}^p(\hat{\kappa}, \kappa)/K$$

The conclusion follows by applying Theorem 5.1 with $p = 1$. □

D Algorithmic Details of Learning with a Wasserstein Loss

In Section 5, we describe the statistical generalization properties of learning with a Wasserstein loss function via empirical risk minimization on a general space of classifiers $\mathcal{H}$. In all the empirical studies presented in the paper, we use the space of linear logistic regression classifiers, defined by

$$\mathcal{H} = \left\{ h_\theta(x) = \left(\frac{\exp(\theta_k^\top x)}{\sum_{j=1}^K \exp(\theta_j^\top x)} \right)_{k=1}^K : \theta_k \in \mathbb{R}^D, k = 1, ..., K \right\}$$

We use stochastic gradient descent with a mini-batch size of 100 samples to optimize the empirical risk, with a standard regularizer $0.0005 \sum_{k=1}^K \|\theta_k\|_2^2$ on the weights. The algorithm is described in Algorithm 2, where WASSERSTEIN is a sub-routine that computes the Wasserstein loss and its subgradient via the dual solution as described in Algorithm 1. We always run the gradient descent for a fixed number of 100,000 iterations for training.

Algorithm 2 SGD Learning of Linear Logistic Model with Wasserstein Loss

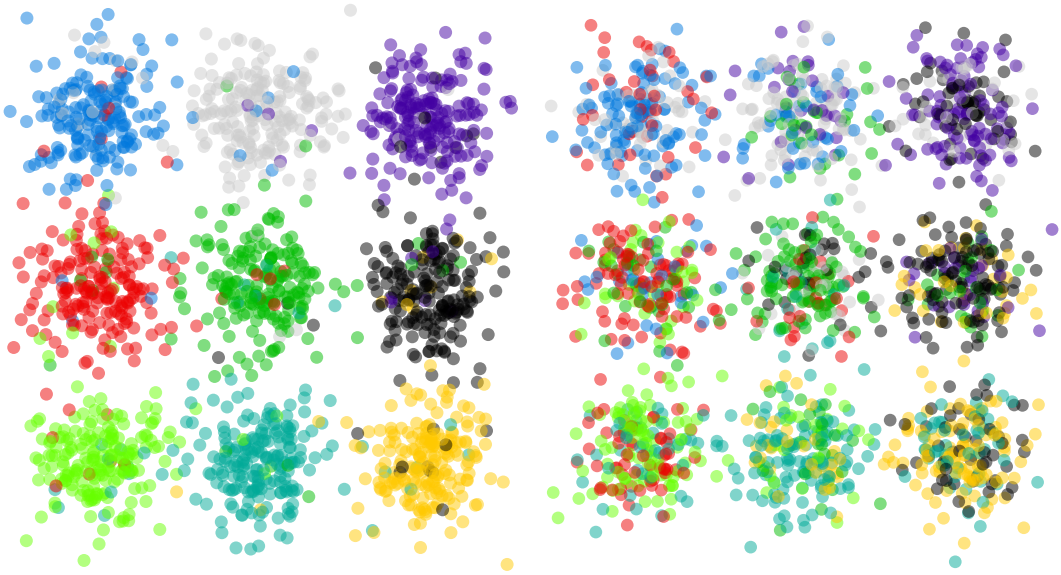
Init θ^1 randomly.
for $t = 1, \ldots, T$ **do**
  Sample mini-batch $\mathcal{D}^t = (x_1, y_1), \ldots, (x_n, y_n)$ from the training set.
  Compute Wasserstein subgradient $\partial W_p^p / \partial h_\theta|_{\theta^t} \leftarrow$ WASSERSTEIN$(\mathcal{D}^t, h_{\theta^t}(\cdot))$.
  Compute parameter subgradient $\partial W_p^p / \partial \theta|_{\theta^t} = (\partial h_\theta / \partial \theta)(\partial W_p^p / \partial h_\theta)|_{\theta^t}$
  Update parameter $\theta^{t+1} \leftarrow \theta^t - \eta_t \partial W_p^p / \partial \theta|_{\theta^t}$
end for

Note that the same training algorithm can easily be extended from training a linear logistic regression model to a multi-layer neural network model, by cascading the chain-rule in the subgradient computation.

E Empirical study

E.1 Noisy label example

We simulate the phenomenon of label noise arising from confusion of semantically similar classes as follows. Consider a multiclass classification problem, in which the labels correspond to the vertices on a $D \times D$ lattice on the 2D plane. The Euclidean distance in $\mathbb{R}^2$ is used to measure the



(a) Noise level 0.1 (b) Noise level 0.5

Figure 8: Illustration of training samples on a 3x3 lattice with different noise levels.

semantic similarity between labels. The observations for each category are samples from an isotropic Gaussian distribution centered at the corresponding vertex. Given a noise level t, we choose with probability t to flip the label for each training sample to one of the neighboring categories[8], chosen uniformly at random. Figure 8 shows the training set for a 3×3 lattice with noise levels $t = 0.1$ and $t = 0.5$, respectively.

Figure 2 is generated as follows. We repeat 10 times for noise levels $t = 0.1, 0.2, \ldots, 0.9$ and $D = 3, 4, \ldots, 7$. We train a multiclass linear logistic regression classifier (as described in section D of the Appendix) using either the standard KL-divergence loss[9] or the proposed Wasserstein loss[10]. The performance is measured by the mean Euclidean distance in the plane between the predicted class and the true class, on the test set. Figure 2 compares the performance of the two loss functions.

E.2 Full figure for the MNIST example

The full version of Figure 4 from Section 6.1 is shown in Figure 9.

E.3 Details of the Flickr tag prediction experiment

From the tags in the Yahoo Flickr Creative Commons dataset, we filtered out those not occurring in the WordNet[11] database, as well those whose dominant lexical category was "noun.location" or "noun.time." We also filtered out by hand nouns referring to geographical location or nationality, proper nouns, numbers, photography-specific vocabulary, and several words not generally descriptive of visual content (such as "annual" and "demo"). From the remainder, the 1000 most frequently occurring tags were used.

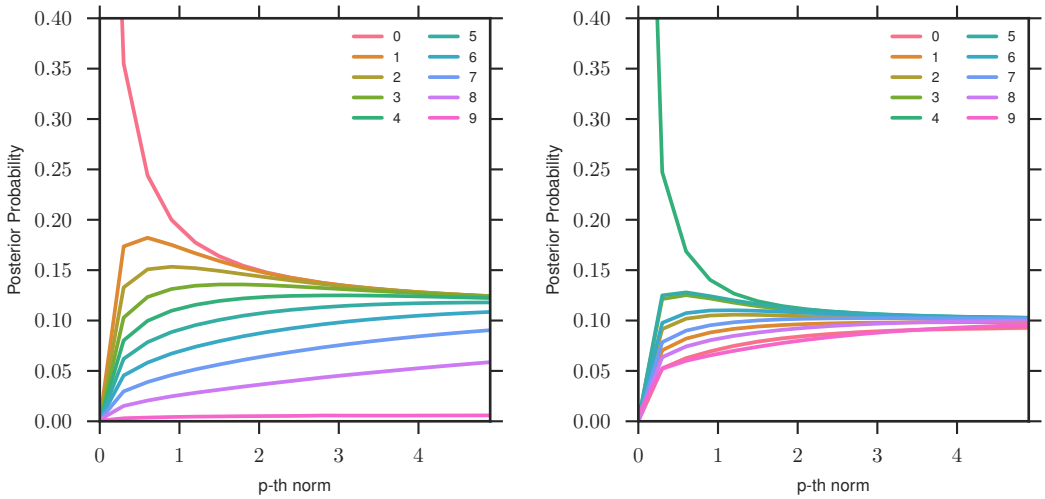
We list some of the 1000 selected tags here. The 50 most frequently occurring tags: *travel, square, wedding, art, flower, music, nature, party, beach, family, people, food, tree, summer, water, concert, winter, sky, snow, street, portrait, architecture, car, live, trip, friend, cat, sign, garden, mountain, bird, sport, light, museum, animal, rock, show, spring, dog, film, blue, green, road, girl, event, red,*

[8] Connected vertices on the lattice are considered neighbors, and the Euclidean distance between neighbors is set to 1.

[9] This corresponds to maximum likelihood estimation of the logistic regression model.

[10] In this special case, this corresponds to weighted maximum likelihood estimation, c.f. Section C.

[11] `http://wordnet.princeton.edu`



(a) Posterior prediction for images of digit 0.

(b) Posterior prediction for images of digit 4.

Figure 9: Each curve is the predicted probability for a target digit from models trained with different p values for the ground metric.

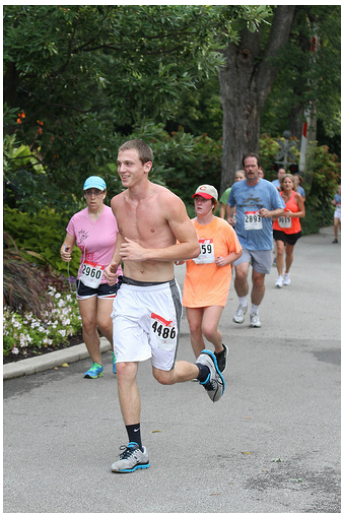
fun, building, new, cloud. ... and the 50 least frequent tags: *arboretum, chick, sightseeing, vineyard, animalia, burlesque, key, flat, whale, swiss, giraffe, floor, peak, contemporary, scooter, society, actor, tomb, fabric, gala, coral, sleeping, lizard, performer, album, body, crew, bathroom, bed, cricket, piano, base, poetry, master, renovation, step, ghost, freight, champion, cartoon, jumping, crochet, gaming, shooting, animation, carving, rocket, infant, drift, hope.*

The complete features and labels can also be downloaded from the project website[12]. We train a multiclass linear logistic regression model with a linear combination of the Wasserstein loss and the KL divergence-based loss. The Wasserstein loss between the prediction and the normalized groundtruth is computed as described in Algorithm 1, using 10 iterations of the Sinkhorn-Knopp algorithm. Based on inspection of the ground metric matrix, we use p-norm with $p = 13$, and set $\lambda = 50$. This ensures that the matrix $\mathbf{K}$ is reasonably sparse, enforcing semantic smoothness only in each local neighborhood. Stochastic gradient descent with a mini-batch size of 100, and momentum 0.7 is run for 100,000 iterations to optimize the objective function on the training set. The baseline is trained under the same setting, using only the KL loss function.

To create the dataset with reduced redundancy, for each image in the training set, we compute the pairwise semantic distance for the groundtruth tags, and cluster them into "equivalent" tag-sets with a threshold of semantic distance 1.3. Within each tag-set, one random tag is selected.

Figure 10 shows more test images and predictions randomly picked from the test set.

[12] http://cbcl.mit.edu/wasserstein/



(a) **Flickr user tags**: zoo, run, mark; **our proposals**: running, summer, fun; **baseline proposals**: running, country, lake.

(b) **Flickr user tags**: travel, architecture, tourism; **our proposals**: sky, roof, building; **baseline proposals**: art, sky, beach.

(c) **Flickr user tags**: spring, race, training; **our proposals**: road, bike, trail; **baseline proposals**: dog, surf, bike.

(d) **Flickr user tags**: family, trip, house; **our proposals**: family, girl, green; **baseline proposals**: woman, tree, family.

(e) **Flickr user tags**: education, weather, cow, agriculture; **our proposals**: girl, people, animal, play; **baseline proposals**: concert, statue, pretty, girl.

(f) **Flickr user tags**: garden, table, gardening; **our proposals**: garden, spring, plant; **baseline proposals**: garden, decoration, plant.

(g) **Flickr user tags**: nature, bird, rescue; **our proposals**: bird, nature, wildlife; **baseline proposals**: ature, bird, baby.

Figure 10: Examples of images in the Flickr dataset. We show the groundtruth tags and as well as tags proposed by our algorithm and baseline.