Improving Optimal Transportation Classification Methods in Machine Learning

Alistair Barton
92543164

Allison Tai
49607104

Abstract

In this project we will adapt optimal transportation methods used for domain adaptation and image retrieval to classification purposes. These methods have the advantage of requiring little to no training, and being robust to noisy data. However it also presents several challenges, such as a sensitivity to outliers and an ignorance of the symmetries of the space. To counter this, we propose and investigate several modifications to our method. These include using different cost functions, creating a model for each class, using optimal transportation to take advantage of symmetries in the data, and convolutions. Modelling in particular is shown to yield a significant improvement, while changing the cost function offers a simple improvement. These methods may easily be applied to domain adaptation and image retrieval in future studies.

1 Introduction

Optimal transportation (OT) is a mathematical technique for comparing two distributions $(\mu, \Omega), (\nu, \Omega)$ according to a distance on their universal set Ω. It does this by computing the minimal 'cost' of transporting all the mass in one distribution to the other, where cost is (often) equal to distance. This method of comparing distributions has been applied to image retrieval by Rubner et al. [2000]—who use this method to compare the distributions of pixels that make up an image—and to domain adaptation by Courty et al. [2014]—who use OT to compare the distribution of samples drawn from two related processes. Our goal in this project is to investigate the application of OT as a classification technique using methods motivated by the two papers cited above (which will be discussed in detail in Section 2).

In addition to the appealing interpretation of OT as a distance between distributions, OT has some strengths relative to alternative classification methods like convolutional neural networks (CNNs). Firstly, as a distance between functions, it requires little to no training—a cross-validation procedure may be helpful to select a regularization parameter, as discussed in Section 3. In addition, OT methods are robust to random noise, which is another weakness of CNNs [Szegedy et al., 2013, §4].

Optimal Transportation Theory *Notation: bold symbols will denote vectors or matrices in their entirety, while when $\boldsymbol{x}$ refers to an image, $[i]$ will indicate the location that the x_i coordinate refers to.*

Optimal transportation is the theory of calculating the minimal 'cost' of transporting mass from one distribution to another. In a discrete setting where the distributions are $\boldsymbol{x} \in \mathbb{R}^m, \boldsymbol{y} \in \mathbb{R}^n$, this may be rephrased as the problem of calculating

$$C(\boldsymbol{x}, \boldsymbol{y}) = \min_{\gamma \geq 0} \left\{ \sum_{i,j} c_{ij}\gamma_{ij} = \langle \boldsymbol{c}, \boldsymbol{\gamma} \rangle_F \,\middle|\, \sum_j \gamma_{ij} = x_i, \sum_i \gamma_{ij} = y_j \right\}, \tag{OT}$$

where $\langle \cdot, \cdot \rangle_F$ indicates the Frobenius inner product and the cost function $c_{ij} \in \mathcal{M}(\mathbb{R}^m, \mathbb{R}^n)$ is the 'cost of transporting' a unit mass from the location indicated by $[i]$ to $[j]$. When the cost function is interpreted as a distance function, this provides a natural 'distance' between distributions that inherits the geometry of the space the distribution is over [Rachev, 1998, §1.4]—for example, the cost of transporting between two dirac masses will be equivalent to the distance between their locations. The applications of OT in machine learning differ in what they choose to be their distributions, and hence geometry.

OT has three major weaknesses that we will focus on:

(1) OT can be sensitive to outliers (shown in Figure 2).

(2) With regards to image retrieval, OT can only compare entire images without taking into account certain symmetries (e.g. of rotation). For example, it would say a slanted 1 is closer to '7' than to '1'.

(3) In image retrieval methods, OT may not focus on the key features of an image.

Another weakness of OT that we will not address here is the relatively slow classification time of $\mathscr{O}(d^2)$ per sample, where d is the number of features [Cuturi, 2013, §5].

To attack problem (1), we propose using cost matrices c_{ij} that are concave functions of the distance between $[i], [j]$ (motivated by Figure 2). We also propose altering the method by creating typical models of our labelled samples rather than computing the transportation cost to a few labelled samples (which may themselves be outliers). For problem (2), we propose using information from the optimal transport plan γ (minimizing Eq. OT) to transform the original sample before running OT once more. Lastly, in an attempt to resolve problem (3), we propose combining OT with a shallow CNN composed of a single convolution layer. Each of these methods will be detailed and motivated in Section 3, and will be tested in Section 4.

2 Related Works

There are two main applications of OT in machine learning based on what the distributions $\boldsymbol{x}$ and $\boldsymbol{y}$ refer to: image retrieval and domain adaptation. We refer here to a couple of foundational papers for these applications to review their methods and how they dealt with the problems mentioned in the previous section.

Image Retrieval The classic application of OT is to the problem of image retrieval pioneered by Rubner et al. [2000]. Here $\boldsymbol{x}$ and $\boldsymbol{y}$ are taken to represent the distribution of pixels associated to different images, and $C(\boldsymbol{x}, \boldsymbol{y})$ may be interpreted as the 'distance' between the images. The convention here is to use c_{ij} to represent the Euclidean distance between the pixels $[i]$ and $[j]$.

Rubner et al. propose using colour distances and Gabor filters in order to focus on colour matching and textures—relevant to problem (3). This is shown to work well for this specific purpose; however, it cannot learn from labelled data and hence is rather rigid for the purposes of supervised learning. The paper also suggests a method of increasing computational speed using signatures, defined as a set of main clusters of a distribution, that we will not investigate here.

Frogner et al. [2015] expands on Rubner's distance metric, applying it to supervised learning by transforming it into a loss function. They do this by proposing a relaxation that extends the regularized OT proposed by previous literature to unnormalized measures, which involved replacing the constraints on the transport marginals with soft penatlies. They then tested their method on MNIST digits classification and Flickr tag prediction. However, although the paper demonstrates a novel method adapting OT to supervised learning, it does not address the weaknesses inherent to OT, for example, the idea that OT does not focus on key features, or that it may have difficulty classifying transformed digits.

Domain Adaptation Courty et al. [2014] proposed that OT be applied to the problem of domain adaptation. Here x and y are taken to be the empirical distribution of samples drawn from different but related distributions—a common example is a photo headshot and photo of the same face in profile. Here, each element of $\boldsymbol{x}$ (called the *source distribution*) and $\boldsymbol{y}$ (the *target distribution*) is an image, ie. $\boldsymbol{x}, \boldsymbol{y}$ are vectors of images. Courty et al. propose using the L^2 distance between images $[i]$ and $[j]$ as the cost function: the root-sum-square of the difference in the pixel intensity.

Courty et al. use group regularization to attack problem (1)—where each group corresponds to a label—however this is only possible in supervised or semi-supervised learning schemes, and even then can be difficult to efficiently solve.

3 Description of Methods

For optimal tranpsortation to be a valid tool, there needs to be a fast method of solving (OT). Despite being a linear program, simplex-like methods are $\mathscr{O}(d^3 \log d)$ [Pele and Werman, 2009, §2.1], and are not practical at the scales of the problems we will be looking at. Cuturi [2013] circumvents (OT) by proposing the use of a regularizing term based on the negentropy of the distribution:

$$C_\lambda(\boldsymbol{x}, \boldsymbol{y}) = \min_{\gamma \geq 0} \left\{ \langle \boldsymbol{c}, \boldsymbol{\gamma} \rangle_F + \lambda^{-1} \sum_{ij} \gamma_{ij} \log \gamma_{ij} \,\middle|\, \sum_j \gamma_{ij} = x_i, \sum_i \gamma_{ij} = y_j \right\}. \tag{ROT}$$

Cuturi then proposes a variant of Sinkhorn's algorithm that minimizes (ROT), which has the added benefit of being able to calculate the costs between a source distribution and many target distributions simultaneously. The regularizing term made be understood as corresponding to the prior assumption that γ_{ij} are drawn from iid poisson distributions (see Appendix A).

A last method for solving OT is presented by Schmitzer [2016], which is a faster method for solving the unregularized problem (OT) taking advantage of some geometric properties of optimal transportation. While certainly worth a closer look, we will neglect this method here on the grounds that (a) Schmitzer only presents an explicit algorithm for costs similar to $c_{ij} = \|[i] - [j]\|^2$ (which we will show is not a favourable cost in Section 3.1), without an obvious extension to other costs, and (b) negentropy regularization appears to improve the accuracy of the algorithm.

We use two different classification functions (that will be specified) inspired from the use of OT in image retrieval (IR) and from domain adaptation (DA):

$$c^i = \arg\min_c C_\lambda(\boldsymbol{x}^i, \boldsymbol{y}^c), \tag{IR}$$

$$c^i = \arg\max_c \gamma_\lambda(\boldsymbol{x}_i, \boldsymbol{y}_c). \tag{DA}$$

IR corresponds to finding the class representative closest to our image $\boldsymbol{x}^i$ under a cost function, while DA finds the representative with the strongest coupling with respect to $\boldsymbol{x}_i$ given the optimal coupling γ_λ between some samples and class representatives. The corresponding 'UGM' model for these classification schemes is shown in Figure 1.

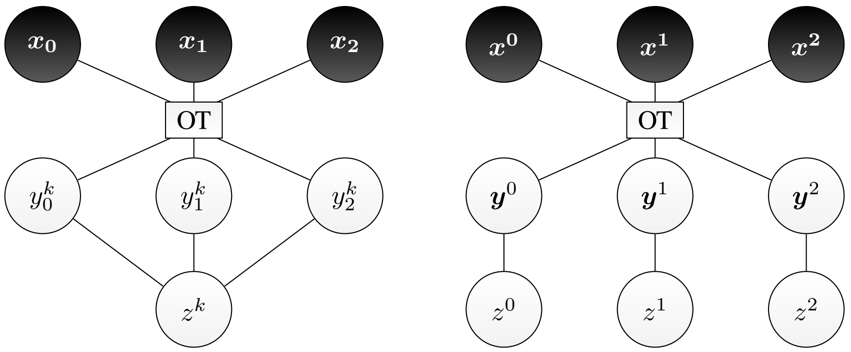

Figure 1: (OT) cannot be represented as a graphical model because it directly relates each of the samples of both distributions. Here the hypergraphical models for classification based on image retrieval (left) and domain adaptation (right) are diagrammed using colour conventions from class. The label z^k corresponding to $\boldsymbol{y}^k$ can be a hidden variable in the unsupervised case or a given variable in the supervised case. [OT] represents the hyperedge imposed by optimal tranpsort connecting all the elements above it with the elements of below it under the potential $-\log \phi(\boldsymbol{x}, \boldsymbol{y}) = C(\boldsymbol{x}, \boldsymbol{y})$.

3.1 Cost Functions

The choice of cost function in (ROT) a non-trivial one. The traditional choice is to use some sort of distance function on the underlying space, in which case (OT) forms a metric on the space of probability distributions, called the Wasserstein distance [Villani, 2003, §7.1].

In image retrieval, (OT) may be understood as the MLE under the probability that each pixel is distributed according to a distribution $p([i] \sim [j]) \propto \exp\left(-c_{ij}\right)$. Heuristically, it follows that our cost function ought to be of the form $c_{ij} = h(\|[i] - [j]\|)$ (ie. translation and rotation invariant) where h is monotonically increasing. What we choose to be h has a big effect as demonstrates in Figure 2: if h is convex, our transportation favours many smaller shifts, while if h is concave our transportation favours larger jumps [Gangbo and McCann, 1996]—in particular, a concave[1] cost will always avoid transporting mass that is in common between the two distributions. This motivates us to examine how convex and concave cost functions affect the performance of OT methods. This experiment may be further motivated by our goal of creating an OT CNN, where methods of computing the gradient of (OT) [Ambrosio et al., 2007] require using the strictly convex $h = (\cdot)^2$. This is further discussed in Appendix B.

3.2 Modelling Samples

Another advantage of optimal transportation methods is that it offers us a method of calculating distances between distributions. Thus far we have not made use of this property—merely taking the distributions to be samples. To take advantage of this interpretation in image retrieval, we can calculate the distance between samples and some set of probability distributions that represent the space of examples—ie. models for our samples.

The question now is how to obtain the best models. Two simple methods come to mind: (1) mixture of Bernoullis, and (2) OT barycenter. We will consider these methods in a supervised setting, but this may be extended to unsupervised learning by using the EM algorithm [Dempster et al., 1977]. For (1) this would result in the exact mixture of Bernoullis method used in assignment 3, whereas for (2) this would result in a version of k-'means' clustering, where the averaging step is replaced with a barycenter step as discussed by Ye et al. [2017].

3.3 γ-based Transformations

Another benefit of optimal transport is that it doesn't just tell us the similarity between two distributions, but the similarity between two pixels given that two distributions correspond to eachother (this

[1] A concave cost refers to the concavity of h.

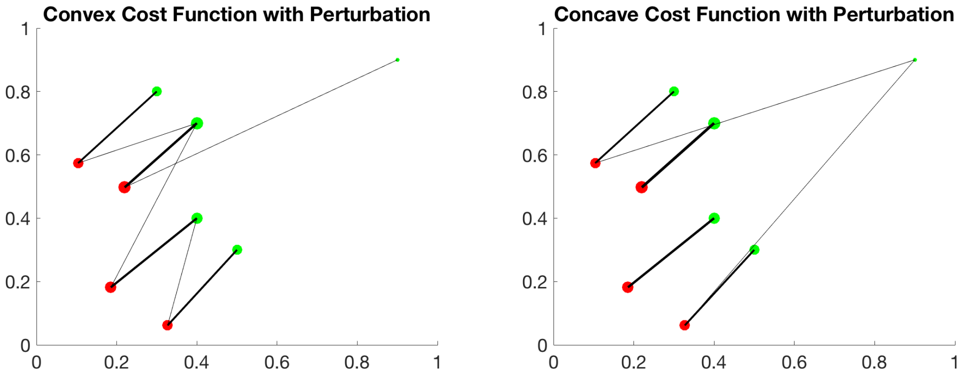

Figure 2: OT between a 2-dimensional dataset in red and a slightly translated version of itself with a small outlier added in green. The line weights correspond to the mass trasported along each conection. The cost used is of the form $c_{ij} = h(\|[i] - [j]\|)$, where h is strictly convex on the left and strictly concave on the right. OT with a concave transportation cost is relatively robust to this outlier, while a convex cost allows its effects to cascade through the system.

is determined by the coupling strength γ_{ij} of the minimizer of (ROT)). Knowing this coupling allows us to apply certain symmetries relevant to the data in order to optimize this coupling.

For example, we may not care whether a certain pattern occurs in an image, only that it appears (that is we have a symmetry[2] of translation). To accomodate this we might first find the optimal transportation between the image and each image in our collection of sample patterns, then apply a translation to each of our patterns. This translation would be determined by the average translation of pixels according to $\boldsymbol{\gamma}$, ie.

$$\sum_{i,j} \gamma_{ij} \left([j] - [i] \right).$$

Now the constraints on (ROT) imply that this is simply translating the two images such that they have the same center of mass, independent of the peculiarities of the images. However, novel methods may be attained by applying this same reasoning to symmetries of scale and rotation. This is what we refer to when we say $\boldsymbol{\gamma}$-based transformation.

Note that this does not completely resolve the symmetry, as would, for example, taking the minimum OT cost over all rotations/scalings of $\boldsymbol{x}$. In particular, if we consider the numbers 6 and 9—which are equivalent if their space is rotationally symmetric—$\boldsymbol{\gamma}$-based rotation would not find them identical. This is because the optimal coupling would associate the top half of each number with the other's, and hence the $\boldsymbol{\gamma}$-based rotation straightens numbers out, rather that flip them around.

3.4 CNN

Lastly, we also attempted to create a shallow (one-layer) convolution neural network for the MNIST dataset. This was pursued by creating a 3×3 filter that was convolved with both the source and target data before the optimal transport step (in the hopes that convolving would yield a more identifiable image).

While OT theory [Ambrosio et al., 2007] does give us a gradient of (OT) under certain costs, we decided not to use this result for several reasons detailed in Appendix B. Rather, we applied coordinate-wise gradient descent using numerical approximations of the partial derivatives. We used a loss function of

$$L(f) = \sum_{i \sim j} \gamma_\lambda^f(\boldsymbol{x}_i, \boldsymbol{y}_j), \tag{3.1}$$

where $i \sim j$ means i and j share the same label and γ_λ^f solves (ROT) with source/target samples convolved by f.

Ultimately this technique proved impractical as the optimal convolution varies significantly depending on the samples used, and to use a sufficiently large sample size requires impractical amounts of time. We shall revisit this in Section 5.

4 Experiments & Analysis

4.1 Cost Functions

Following the intuition in Section 3.1, a concave cost would be advantageous when we have outlying data points (a convex cost might be forced shift these 'excess' data points onto a nearby point, leading to a chain reaction).

To test this, we experimented with the use of different cost functions, specifically using DA classification (this allows us to better control the number of outliers, which here correspond to a different number of samples from a class in the sample and test set). Following DA methods, we took the source domain to be a sample set, and our target domain to be a set of labelled examples. We then solved (ROT) using concave and convex cost functions, and applied DA classification.

To test the effects of differing sample/target distributions, we first generated 3 random source samples for each label (0:9), then randomly generated a number of additional source and target samples

[2] Note that when we say symmetry, we refer to the symmetry of the space—ie. a rotational symmetry means a square and a diamond are equivalent in our considerations—and not to symmetries of the image itself.

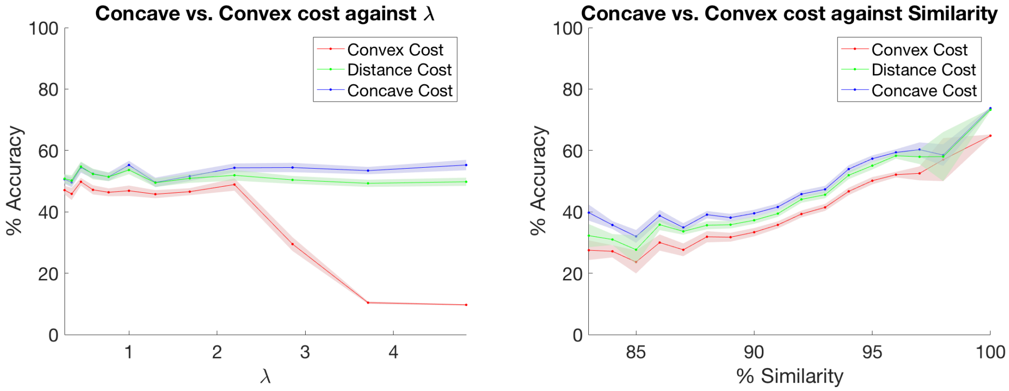

Figure 3: We compare the performance of costs that that corresponds to the L^2 norm ($c_{ij} = \|[i]-[j]\|_2$) with the strictly convex cost $\|[i]-[j]\|_2^2$ and the strictly concave cost $\log(1+\|[i]-[j]\|_2)$ (other concave/convex costs perform similarly). Mean performance and standard error are shown. On the left we compare the performance against the value of the regularizing λ used ($\sim 95\%$ similarity) while on the right we see how the costs compare as the similarity between x and y is increased ($\lambda = 1$). At all points the concave cost performs at least as well as the distance cost and appears to be more robust.

with random labels. We then calculated the $\%$ similarity by counting the percent of the source labels that were not represented in the target data. Running this program many times resulted in the graphs shown in Figure 3, which compares the accuracy for the cost function that is the L^2 distance $d_{ij} := \left\|\boldsymbol{x}_i - \boldsymbol{y}_j\right\|_2$ along with the strictly concave cost $\log(1+d_{ij})$ and the strictly convex cost d_{ij}^2. We also examined the effects of different λ which illustrates the convergence of these costs as $\lambda \searrow 0$ (as the regularizing term becomes dominant).

We can see that our intuition is accurate: in general concave costs are superior. In particular the strictly convex cost is no better than random guessing for large values of λ. There is also a small but consistent improvement in using a strictly concave cost in terms of robustness with respect to λ and similarity. This is characteristic of other strictly concave cost functions that we tried (eg. atan(d_{ij}), atanh(d_{ij})).

4.2 Modelling Samples

To test our modelling of samples we compare the similarity between a given image from the MNIST dataset $\boldsymbol{x}$ and a collection of models $\boldsymbol{y}_i$ using (ROT). Note that this is much weaker and less sophisticated than the method discussed by Rubner et al. [2000], who use a color distance in their metric. We obtained the prototypical models in two different ways: (1) using samples taking from MNIST corresponding to each number, (2) fitting a mixture of Bernoulli model to each number[3]. Once we had our samples, we used the IR classification scheme and tracked their accuracy.

The Bernoulli method adapted the code from assignment 3 that applied mixture of Bernoullis to all of MNIST data. The only difference is that we applied this function to 5000 random samples corresponding to each number (along with decreasing the convergence criterion). Both of these methods are able to generate multiple models, and their success rate for different numbers of models is shown in Table 1.

Our results clearly show that obtaining a mixture of Bernoullis model offers a consistent and significant improvement over comparing with random samples from each class.

4.3 γ-based Transformations

Motivated by Section 3.3, we investigated how translating and scaling images can negatively impact classification accuracy with OT, and how we can mitigate such drops in accuracy by implementing aforementioned γ-based tranformation. This involved solving (OT) between a distorted test image

[3] We did not manage to get the barycentric method to work within the time constraints

Table 1: Classification accuracy of different models, either sampling from MNIST, or fitting mixture of Bernoullis to a subset of each labelled dataset.

	Classification accuracy (%)					
	Number of models					
	1	2	3	4	5	6
MNIST sampling	56	61	69	67	65	57
Mixture of Bernoullis	70	79	77	89	86	92

and each of our training samples, modeled with mixtures of Bernoullis as outlined in Section 4.2. We then translated the test example by the average distance each pixel was being translated according to γ_λ. We then re-ran standard ROT on the translated image to find the best IR classification.

To experiment, we generated test sets using the MNIST dataset: the first test set represented a translated version of all the digits, and the second a scaled version. We created the former by shifting each digit a random number of pixels, while we generated the latter by either compressing or expanding each image, but maintained the 28 by 28 pixel size. With both, transformation along each axis was handled independently. In all cases, we constrained the values of translation and scaling to avoid distorting the digit too much, wanting to maintain recognizability. A selection from the datasets we created can be seen in Figure 4.

We then ran OT with our Bernoulli-based EMD method, to test how these transformations affected classification accuracy, for a total of 200 classifications each. We summarize our results in Table 2. Notably, OT seems very robust to translations, as the classfication success rate of the translated MNIST dataset is not consistently positive, compared to what we saw in Section 4.2. However, this may be due to the fact that the models for each class share similar centre of masses, hence no class obtains a significant advantage from the pre-translated samples. Despite this, we still implemented γ-based translations as a proof of concept that we can recover from distortions in imagesets by using coupling strength γ.

When we tried to apply a similar idea to the scaled dataset, we discovered that OT computed using Sinkhorn's algorithm often fails to converge, which prevents us from reliably applying transformations. Further investigation is needed to understand why, so we leave γ-based scaling to future work.

Table 2: Classification accuracy of the mixture of Bernoullis method on either translated (T) or scaled (S) MNIST digits, with or without the matching γ-transformation.

	Classification accuracy (%)					
	Number of models					
	1	2	3	4	5	6
T, no transformation	66	75	79	80	82	88
T, γ-based transformation	60	70	74	83	88	86
S, no transformation	failed to converge					

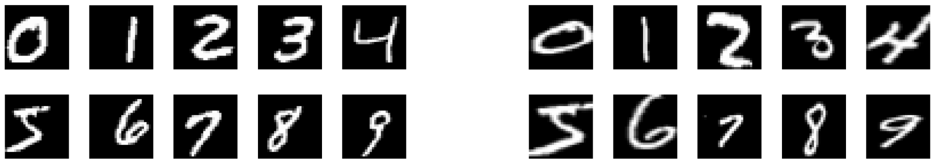

Figure 4: Some selected figures from the two versions of transformed MNIST digit datasets we generated. On the left, we have some examples from our translated dataset. On the right are examples from our scaled dataset.

5 Discussion & Future Work

In this project we examined several methods of improving OT methods in machine learning. We did this by first interpreting the two major applications (domain adaptation and image retrieval) as classification problems and examing the change in their success rates with the implementation of some additional tools.

Among these tools, we found that using concave or strictly concave cost functions is preferable to convex cost functions, with strictly concave costs giving a consistent $1 - 5$ percentage point improvement over convex costs. We also showed how replacing the test samples with models generated by mixture of Bernoullis can improve the classification by 10-30 percentage points. We discussed how OT might be used to quickly address 'small' symmetries in our space, however we found that matching center of mass did not have a consistent positive effect and were unable to apply the method to scaled images. Lastly, we investigated the possibility of combining OT with a single layer CNN, and showed that this is not practical using traditional methods (alternative methods are proposed below).

We consciously decided to focus on breadth rather than depth for this project. As a result, we see the primary weakness in our work to be a lack of deeper testing (ie. on other datasets other than MNIST). This would certainly have been a priority had we some extra time after the completion of our project. We see the strength of our contribution to be the wide variety of fairly generalizable, simple techniques of extending OT methods that we tested, and in particular the improvement generated by Bernoulli modelling. Another weakness is the amount of work that we left unfinished (in our eyes), discussed in the following paragraph.

Future Work While this project took on many different dimensions, we see many avenues for future work, that we would've liked to explore had we more time—in addition to the barycentric models we mentioned in Section 3.2.

In Section 3.3, we mentioned how rotation symmetries could be countered using a γ-based rotation. Time constraints prevented us from investigating these methods, but we think it would have been interesting to investigate these in detail. One question that we would like to answer is when will γ-based rotations resolve rotational symmetry (ie. match samples with a rotated version of themselves)? In cases like '6' versus '9' the answer seems simple (when the angle of rotation is $\lesssim 90°$), however in cases where our class samples consist of 'Y' and 'L' the answer is not obvious.

We also do not believe that our inability to implement OT CNNs in Section 3.4 reflects their impracticality. Reducing dimensionality—by using ReLu layers or a larger convolution stride—would speed up the process for larger sample sizes. Potential results from OT theory could also help: a gradient result for concave costs, or a method of approximating the Lipschitz constant with respect to different coordinates, or even a method of simultaneously computing multiple transportation costs.

References

L. Ambrosio, G. Savare, and L. Zambotti. Existence and Stability for Fokker-Planck equations with log-concave reference measure. *ArXiv e-prints*, April 2007.

B. J. Brewer and M. J. Francis. Entropic Priors and Bayesian Model Selection. In P. M. Goggans and C.-Y. Chan, editors, *American Institute of Physics Conference Series*, volume 1193 of *American Institute of Physics Conference Series*, pages 179–186, December 2009. doi: 10.1063/1.3275612.

N. Courty, R. Flamary, and D. Tuia. Domain adaptation with regularized optimal transport. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2014, Nancy, France, September 15-19, 2014. Proceedings, Part I*, pages 274–289, 2014. ISBN 978-3-662-44848-9. doi: 10.1007/978-3-662-44848-9_18. URL `http://dx.doi.org/10.1007/978-3-662-44848-9_18`.

M. Cuturi. Sinkhorn Distances: Lightspeed Computation of Optimal Transportation Distances. *Neural Information Processing Systems (NIPS)*, page 2292–2300, 2013. URL `https://papers.nips.cc/paper/4927-sinkhorn-distances-lightspeed-computation-of-optimal-transport.pdf`.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977. ISSN 00359246. URL `http://www.jstor.org/stable/2984875`.

C. Frogner, C. Zhang, H. Mobahi, M. Araya, and T. A. Poggio. Learning with a Wasserstein Loss. In *Advances in Neural Information Processing Systems 28*, pages 2053–2061. Curran Associates, Inc., 2015. URL `http://papers.nips.cc/paper/5679-learning-with-a-wasserstein-loss.pdf`.

W. Gangbo and R. J. McCann. The geometry of optimal transportation. *Acta Math.*, 177(2):113–161, 1996. doi: 10.1007/BF02392620. URL `http://dx.doi.org/10.1007/BF02392620`.

E. T. Jaynes and G. L. Bretthorst. *Probability Theory: The Logic of Science*. Cambridge University Press, 2003. ISBN 9780521592710. URL `https://books.google.ca/books?id=tTN4HuUNXjgC`.

O. Pele and M. Werman. Fast and robust earth mover's distances. In *ICCV*, 2009.

S. T. Rachev. *Mass transportation problems*. Springer, 1998. ISBN 978-0-387-98350-9.

Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000. ISSN 1573-1405. doi: 10.1023/A:1026543900054. URL `https://www.cs.cmu.edu/~efros/courses/LBMV07/Papers/rubner-jcviu-00.pdf`.

B. Schmitzer. A sparse multiscale algorithm for dense optimal transport. *Journal of Mathematical Imaging and Vision*, 56(2):238–259, 2016. ISSN 1573-7683. doi: 10.1007/s10851-016-0653-9. URL `http://dx.doi.org/10.1007/s10851-016-0653-9`.

C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

C. Villani. *Topics in Optimal Transportation*. Graduate studies in mathematics. American Mathematical Society, 2003. ISBN 9780821833124.

J. Ye, P. Wu, J. Z. Wang, and J. Li. Fast discrete distribution clustering using wasserstein barycenter with sparse support. *IEEE Transactions on Signal Processing*, 65(9):2317–2332, May 2017. ISSN 1053-587X. doi: 10.1109/TSP.2017.2659647.

A MAP Justification for Negentropy Regularization

To our knowledge (eg. [Brewer and Francis, 2009, §2.1]) the current justification for negentropy regularizers is an appeals to Jaynes' principle of entropy maximization [Jaynes and Bretthorst, 2003, §9.7]. We will show how KL divergence regularizer (of which negentropy is just one example) may be in fact obtained from a Maximum a Posteriori estimation.

We make the prior assumption that the couplings γ_{ij} are drawn from independent Poisson distributions of mean $\alpha_{ij} > 0$, where we replace the factorial on the denomonator with Stirling's approximation to get the continuous analog:

$$\begin{aligned} p(\boldsymbol{\gamma}|\boldsymbol{\alpha}) &\propto \prod_{ij} \frac{\alpha_{ij}^{\gamma_{ij}}}{\gamma_{ij}^{\gamma_{ij}} e^{-\gamma_{ij}}} \\ \log p(\boldsymbol{\gamma}|\boldsymbol{\alpha}) &= C_\alpha + \sum_{ij} \gamma_{ij} \left(\log \left(\frac{\alpha_{ij}}{\gamma_{ij}} \right) + 1 \right), \end{aligned}$$

where C_α is a normalizing constant. The key step is to realize that the minimizing γ is invariant under the addition of linear terms of γ_{ij}, as $\sum_{ij} \gamma_{ij} = 1$. This means

$$\arg\max_{\gamma} \log p(\boldsymbol{\gamma}|\boldsymbol{\alpha}) = \arg\min_{\gamma} \sum_{ij} \gamma_{ij} \log \left(\frac{\gamma_{ij}}{\alpha_{ij}} \right) = \arg\min_{\gamma} \mathrm{KL}(\boldsymbol{\gamma} \| \boldsymbol{\alpha}). \tag{A.1}$$

Note that the λ factor can arise by scaling the probability distribution $\gamma \to \lambda\gamma$:

$$\begin{aligned} \log p(\lambda\boldsymbol{\gamma}|\boldsymbol{\alpha}) &= C_\alpha + \sum_{ij} \lambda\gamma_{ij} \left(\log \left(\frac{\alpha_{ij}}{\gamma_{ij}} \right) + 1 - \log\lambda \right), \\ \arg\max_{\gamma} \log p(\lambda\boldsymbol{\gamma}|\boldsymbol{\alpha}) &= \arg\min_{\gamma} \lambda \mathrm{KL}(\boldsymbol{\gamma} \| \boldsymbol{\alpha}). \end{aligned} \tag{A.2}$$

Of course, the assumption that γ_{ij} are independent is naïve: many sample $\boldsymbol{\gamma}$ drawn from such distribution will break the conditions of (OT)—they may not even be normalized!—however the average discrepancy will shrink as the length of our vectors increases (by the law of large numbers).

B Exact OT Gradient for CNNs

Ambrosio et al. [2007] describe how to calculate the derivative of the Wasserstein squared metric as the underlying probability distribution changes. Using our notation, this is $\partial_t C(x_t, y)$ with $c_{ij} = \|[i] - [j]\|_2^2$ and $t \mapsto x_t$ is a continuous path in the space of probability distributions.

This implementation would present a couple of challenges. Firstly, the derivative Ambrosio et al. [2007] propose only applies to OT, rather than ROT, meaning this would require implementing an algorithm for solving (OT) such as the one developed by Schmitzer [2016]. A secondary complication is that the continuity condition of Ambrosio's work would constrain our convolution matrices to be 3×3 stochastic matrices obeying certain properties (at least for the step before applying OT), complicating the gradient descent steps. Of course, as explored in Section 4.1, the L^2-squared cost is not ideal for our purposes.

While the result of Ambrosio et al. is promising (and may merit a look all the same), a similar result that applies to a non-strictly convex cost would be much more beneficial to our work.