A Free-Energy Principle for Representation Learning

Abstract

We employ a formal connection of machine learning with thermodynamics to characterize the quality of learnt representations for transfer learning. We discuss how information-theoretic functionals such as rate, distortion and classification loss of a model lie on a convex, so-called equilibrium surface. We prescribe dynamical processes to traverse this surface under constraints, e.g., an iso-classification process that trades off rate and distortion to keep the classification loss unchanged. We demonstrate how this process can be used for transferring representations from a source dataset to a target dataset while keeping the classification loss constant. Experimental validation of the theoretical results is provided on standard image-classification datasets.

1 Introduction

Information theory creates a lossless representation with the goal of getting the original data back after decoding. Similarly, given images and their labels one could learn a representation with the goal of predicting the correct labels. If this representation is minimal it would discard information in the data that is not correlated with the labels. This makes it unique to the chosen task, it would perform poorly to predict other labels that rely on the discarded information. If instead the representation were to retain redundant information about the data, it could predict other labels correlated with this information. We would like to characterize the discarded information in order to learn representations that can be transferred to other tasks.

Our main idea is to choose a canonical task—in this paper, we pick reconstruction of the original data—to measure the discarded information. Although one can use any task, reconstruction is special because achieving perfect reconstruction entails that the representation is lossless. Information discarded is therefore readily measured as the one that helps improve reconstruction. This leads to the study of the Lagrangian

$$F(\lambda,\gamma) = \min_{\theta\in\Theta,\ e_\theta(z|x),m_\theta(z),d_\theta(x|z),c_\theta(y|z)} \Big\{R + \lambda D + \gamma C\Big\}$$

where rate R is an upper bound on the mutual information z learnt by the encoder $e_\theta(z|x)$ and data x, distortion D measures the quality of reconstruction of the decoder $d_\theta(x|z)$ and C measures the classification loss of classifier $c_\theta(y|z)$.

Contributions. (i) We observe that $F(\lambda,\gamma)$ can be interpreted as a free-energy and corresponds to an "equilibrium surface" of information-theoretic functionals R, D and C. We prove that the equilibrium surface is convex and its dual, the free-energy $F(\lambda,\gamma)$, is concave. (ii) We design stochastic processes to keep the model parameters θ on the equilibrium surface and travel to any feasible values of (R, D, C). We focus on iso-classification process which automatically trades off the rate and distortion to keep the classification loss constant. (iii) We prescribe a dynamical process that allows for controlled transfer of representations. It adapts the model parameters as the task is changed while keeping the classification loss constant.

2 Theoretical setup

$$D = \mathop{\mathbb{E}}_{x\sim p(x)}\left[-\int \mathrm{d}z\ e(z|x)\log d(x|z)\right],\quad R = \mathop{\mathbb{E}}_{x\sim p(x)}\left[\int \mathrm{d}z\ e(z|x)\log\frac{e(z|x)}{m(z)}\right] \tag{1}$$

Consider an encoder $e(z|x)$ that encodes data x into a latent code z and a decoder $d(x|z)$ that decodes z back into the original data x. If the true distribution of the data is $p(x)$ we may define the following functionals. The distortion D measures the quality of the reconstruction through its log-likelihood. The rate

R is a Kullback-Leibler (KL) divergence; it measures the average excess bits used to encode samples from $e(z|x)$ using a code that was built for our approximation of the true marginal on the latent factors $m(z)$. The Shannon entropy is defined as $H = -\mathbb{E}_{x\sim p(x)}[\log p(x)]$.

The functionals in (1) come together to give the inequality $H - D \leq I_e(x;z) \leq R$ where $I_e = \mathrm{KL}(e(z|x) \,||\, p(z|x))$ is the KL-divergence between the learnt encoder and the true (unknown) conditional of the latent factors. For finite capacity variational families, say parameterized by θ, which we denote by $e_\theta(z|x)$, $d_\theta(x|z)$ and $m_\theta(z)$ respectively, one obtains an RD curve (see Fig. 1) corresponding to the Lagrangian [1]

$$F(\lambda) = \min_{e_\theta(z|x), m_\theta(z), d_\theta(x|z)} R + \lambda D. \tag{2}$$

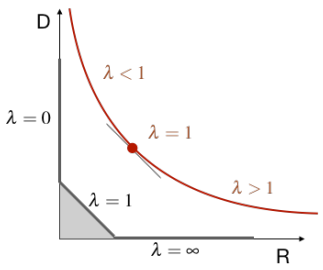


Figure 1: The RD equilibrium surface for infinite capacity (black) and finite capacity (red). The Lagrangian $F(\lambda)$ controls which part the solution of (2) lies in.

This Lagrangian is the relaxation of the idea that given a fixed variational family and data distribution $p(x)$, there exists an optimal value of, say, rate $R = f(D)$ that best sandwiches the above inequality. Let us create a classifier that uses the learnt representation z as the input and set the classification loss to be $C = \mathbb{E}_{x\sim p(x)}\left[-\int \mathrm{d}z\, e(z|x) \log c(y|z)\right]$. We can again consider a Lagrange relaxation of this surface given by

$$F(\lambda, \gamma) = \min_{e(z|x), m(z), d(x|z), c(y|z)} R + \lambda D + \gamma C \tag{3}$$

and obtain the following lemma.

Lemma 1. *The constraint surface $f(R, D, C) = 0$ is convex and the Lagrangian $F(\lambda, \gamma)$ is concave.*

We assume in this paper that the labels are a deterministic function of the data, i.e., $p(y|x) = \delta(y - y_x)$ where y_x is the true label of the datum x. We can solve the variational problem in (3) to get $e(z|x)$ in terms of the other quantities $e(z|x) = \frac{m_\theta(z) d_\theta(x|z)^\lambda c_\theta(y_x|z)^\gamma}{Z_{\theta,x}}$ where the normalization constant is $Z_{\theta,x} = \int \mathrm{d}z\, m_\theta(z) d_\theta(x|z)^\lambda c_\theta(y_x|z)^\gamma$. The objective $F(\lambda, \gamma)$ can now be rewritten as

$$F(\lambda, \gamma) = \min_{\theta\in\Theta} -\langle \log Z_{\theta,x} \rangle_{p(x)}. \tag{4}$$

The surface of constraints $f(R, D, C) = 0$ is called the equilibrium surface because if we define

$$H(z; x, \theta, \lambda, \gamma) \equiv -\log m_\theta(z) - \lambda \log d_\theta(x|z) - \gamma \log c_\theta(y|z) \tag{5}$$

as the Hamiltonian and minimize $\mathbb{E}_{x\sim p(x)}\{\int \mathrm{d}z\, e_{\theta^k}(z|x) H(z; x, \theta^k, \lambda, \gamma)\}$ over a finite dataset with stochastic gradient-based updates, the posterior distribution of the model parameters converges to the Gibbs distribution $p(\theta \mid \text{data}) \propto \exp\left(-2(R + \lambda D + \gamma C)/\sigma\right)$ where $\sigma > 0$ is the step-size [2].

3 Dynamical processes on the equilibrium surface

For any parameters $\theta \in \Theta$, not necessarily on the equilibrium surface, let us define $J(\theta, \lambda, \gamma) = -\langle \log Z_{\theta,x} \rangle_{p(x)}$. If $\theta \in \Theta_{\lambda,\gamma} = \{\theta : -\langle \log Z_{\theta,x} \rangle_{p(x)} = F(\lambda, \gamma)\}$ we have $J(\theta, \lambda, \gamma) = F(\lambda, \gamma)$ which implies $\nabla_\theta J(\theta, \lambda, \gamma) = 0$ for all $\theta \in \Theta_{\lambda,\gamma}$. We are interested in evolving (λ, γ) slowly and simultaneously keeping the model parameters θ on the equilibrium surface; the constraint thus holds at each time instant. The equilibrium surface is parameterized by R, D and C so changing (λ, γ) adapts the three functionals to track their optimal values.

Let us choose some values $(\dot{\lambda}, \dot{\gamma})$ and the trivial dynamics $\frac{\mathrm{d}}{\mathrm{d}t}\lambda = \dot{\lambda}$ and $\frac{\mathrm{d}}{\mathrm{d}t}\gamma = \dot{\gamma}$. The quasi-static constraint leads to the following partial differential equation (PDE)

$$0 \equiv \frac{\mathrm{d}}{\mathrm{d}t}\nabla_\theta J(\theta, \lambda, \gamma) = \nabla_\theta^2 J\, \dot{\theta} + \dot{\lambda}\frac{\partial}{\partial \lambda}\nabla_\theta J + \dot{\gamma}\frac{\partial}{\partial \gamma}\nabla_\theta J \tag{6}$$

valid all $\theta \in \Theta_{\lambda,\gamma}$. At each location $\theta \in \Theta_{\lambda,\gamma}$ the above PDE indicates how the parameters should evolve upon changing (λ, γ). We can rewrite the PDE using the Hamiltonian H in (5) as shown next.

Lemma 2. *Given $(\dot{\lambda}, \dot{\gamma})$, the parameters $\theta \in \Theta_{\lambda,\gamma}$ evolve as $\dot{\theta} = -A^{-1}b_\lambda\, \dot{\lambda} - A^{-1}b_\gamma$ and $\dot{\gamma} = \theta_\lambda \dot{\lambda} + \theta_\gamma \dot{\gamma}$ where H is the Hamiltonian in (5) and the other quantities are given in Appendix E.*

3.1 Iso-classification process

An iso-thermal process in thermodynamics is a quasi-static process where a system remains in thermal equilibrium with its surroundings. We can analogously define an process that adapts parameters of the model θ while the free-energy is subject to slow changes in (λ, γ) but keeps the classification loss constant. We simply add a constraint of the form $\frac{\mathrm{d}}{\mathrm{d}t}C = 0$ in addition to the quasi-static condition $\frac{\mathrm{d}}{\mathrm{d}t}\nabla_\theta J = 0$. This leads to the constrained dynamics (see Appendix F)

$$0 = C_\lambda \dot{\lambda} + C_\gamma \dot{\gamma}; \qquad \dot{\theta} = \theta_\lambda \dot{\lambda} + \theta_\gamma \dot{\gamma}. \tag{7}$$

Observe that we are not free to pick any values for $(\dot{\lambda}, \dot{\gamma})$ for the iso-classification process anymore, the constraint $\frac{\mathrm{d}C}{\mathrm{d}t} = 0$ ties the two rates together. The first constraint in (7) allows us to choose

$$\dot{\lambda} = -\alpha\frac{\partial C}{\partial \gamma} = -\alpha\frac{\partial^2 F}{\partial \gamma^2}; \qquad \dot{\gamma} = \alpha\frac{\partial C}{\partial \lambda} = \alpha\frac{\partial^2 F}{\partial \lambda \partial \gamma} \tag{8}$$

where α is a parameter to scale time. The second equalities in both rows follow because $F(\lambda, \gamma)$ is the optimal free-energy which implies relations like $D = \frac{\partial F}{\partial \lambda}$ and $C = \frac{\partial F}{\partial \gamma}$. We can now compute the two derivatives in (8) using finite differences to implement an iso-classification process.

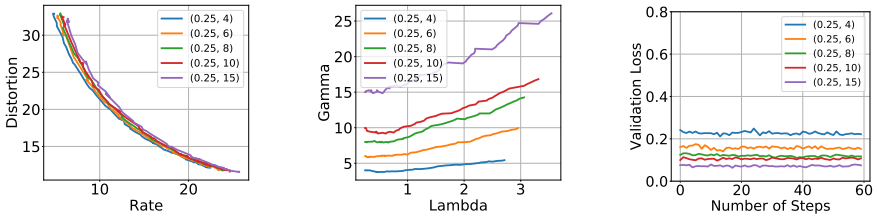
3.2 Iso-classification process with a changing data distribution

The equations (8) show how to adapt the model under perturbations of (λ, γ) to keep the classification error constant. We now discuss a different kind of perturbation, namely the one where the underlying task changes. If i.i.d samples from the source task are denoted by $X^s = \left\{x_1^s, \ldots, x_{n_s}^s\right\}$ and those of the target distribution are $X^t = \left\{x_1^t, \ldots, x_{n_t}^t\right\}$ the empirical source and target distributions can be written as $p^s(x) = \frac{1}{n_s}\sum_{i=1}^{n_s}$ and $\frac{1}{n_t}\sum_{i=1}^{n_t}\delta_{x-x_i^t}$ respectively. For any $t \in [0, 1]$ we interpolate between the two distributions using a mixture $p(x, t) = (1-t)p^s(x) + tp^t(x)$. We can also use techniques from optimal transportation [3] to obtain a better transport (Appendix I); the same dynamical equations given below remain valid. The equilibrium surface $\Theta_{\lambda,\gamma}$ is a function of the task and also evolves with the task. The dynamical process in Lemma 3 keeps the model parameters in equilibrium as the task evolves quasi-statically.

Lemma 3. *Given $(\dot{\lambda}, \dot{\gamma})$, the evolution of model parameters θ for a changing data distribution $p(x, t) = (1-t)p^s(x) + tp^t(x)$ for $t \in [0, 1]$ is*

$$\dot{\theta} = \theta_\lambda \dot{\lambda} + \theta_\gamma \dot{\gamma} + \theta_t. \tag{9}$$

where $\theta_t = -A^{-1}\int \frac{\partial p(x,t)}{\partial t}\, \langle \nabla_\theta H \rangle\, \mathrm{d}x$ and other quantities are defined in Appendix E with the only change that expectations on data x are taken with respect to $p(x, t)$ instead of $p(x)$.



(a) (b) (c)

Figure 2: **Iso-classification process for MNIST.** We run 5 different experiments for initial Lagrange multipliers given by $\lambda = 0.25$ and $\gamma \in \{4, 6, 8, 10, 15\}$. During each experiment, we modify Lagrange multipliers (Fig. 2b) to keep the classification loss constant and plot the rate-distortion curve (Fig. 2a) and the validation loss (Fig. 2c). Validation accuracy is constant for each experiment; it is between 92–98% for these initial values of (λ, γ). Similarly the training loss is almost unchanged during each experiment and takes values between 0.06–0.2 for different values of (λ, γ).

We can now perform an analogous computation as that in Section 3.1 to get the dynamical equations

$$\dot{\theta} = \theta_\lambda \dot{\lambda} + \theta_\gamma \dot{\gamma} + \theta_t; \qquad 0 = C_\lambda \dot{\lambda} + C_\gamma \dot{\gamma} + C_t; \tag{10}$$

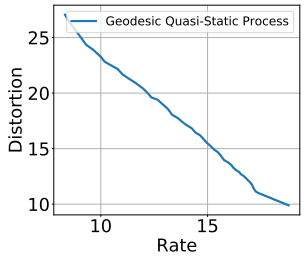
for the iso-classification process with a changing data distribution; expressions for C_λ, C_γ and C_t are given in Appendices F and G.

4 Experimental validation

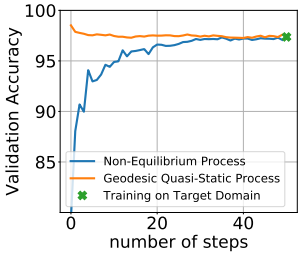
We use the MNIST [4] and CIFAR-10 [5] datasets for our experiments. We show a few representative experiments here; see Appendices B and C for more details.

Iso-classification process. Given a value of the Lagrange multipliers (λ, γ) we first find a model on the equilibrium surface by training from scratch for 120 epochs with the Adam optimizer [6]; the learning rate is set to 10^{-3} and drops by a factor of 10 every 50 epochs. We then run the iso-classification process discussed in Section 3.1. We modify (λ, γ) according to the equations $\dot{\lambda} = -\alpha \frac{\partial C}{\partial \gamma}$ and $\dot{\gamma} = \alpha \frac{\partial C}{\partial \lambda}$ while adapting the model parameters θ to keep them on the dynamically changing equilibrium surface by taking stochastic gradient updates to minimize $J(\lambda, \gamma)$ with a learning rate schedule that looks like a sharp quick increase from zero and then a slow annealing back to zero (see Fig. 5). Fig. 2 shows the result for the iso-classification process for MNIST and Fig. 4 shows a similar result for CIFAR-10.

Iso-classification transfer. We pick the source dataset to be all images corresponding to digits 0–4 in MNIST and the target dataset is its complement, images of digits 5–9. We run the geodesic transfer dynamics from Appendix H and the results are shown in Fig. 3. Fig. 3a shows the variation of rate and distortion during the transfer; as discussed in Appendix H we maintain a constant $\mathrm{d}R/\mathrm{d}D$ during the transfer; the rate decreases and the distortion increases. Fig. 3b shows the validation accuracy during the transfer. The orange curve corresponds to geodesic iso-classification transfer; the blue curve is the result of directly fine-tuning the source model on the target data (note the very low accuracy at the start); the green point is the accuracy of training on the target task from scratch. Results for a similar experiment for transferring between a source dataset that consists of all vehicles in CIFAR-10 to a target dataset that consists of all animals are in Appendix J.



(a)



(b)

Figure 3: **Transferring from source dataset of MNIST digits 0–4 to the target dataset consisting of digits 5–9.**

References

[1] Alexander A Alemi, Ben Poole, Ian Fischer, Joshua V Dillon, Rif A Saurous, and Kevin Murphy. Fixing a broken elbo. *arXiv preprint arXiv:1711.00464*, 2017.

[2] Pratik Chaudhari and Stefano Soatto. Stochastic gradient descent performs variational inference, converges to limit cycles for deep networks. *arXiv preprint arXiv:1710.11029*, 2017.

[3] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.

[4] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[5] A. Krizhevsky. Learning multiple layers of features from tiny images. Master's thesis, Computer Science, University of Toronto, 2009.

[6] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv:1412.6980*, 2014.

[7] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

[8] Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information. *arXiv:1703.00810*, 2017.

[9] Alessandro Achille and Stefano Soatto. On the emergence of invariance and disentangling in deep representations. *arXiv:1706.01350*, 2017.

[10] Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. *arXiv:1612.00410*, 2016.

[11] I Higgins, L Matthey, A Pal, C Burgess, X Glorot, M Botvinick, S Mohamed, and Lerchner A. beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework . In *ICLR*, 2017.

[12] David McAllester. A pac-bayesian tutorial with a dropout bound. *arXiv:1307.2118*, 2013.

[13] Gintare Karolina Dziugaite and Daniel M Roy. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. *arXiv preprint arXiv:1703.11008*, 2017.

[14] Pratik Chaudhari, Anna Choromanska, Stefano Soatto, Yann LeCun, Carlo Baldassi, Christian Borgs, Jennifer Chayes, Levent Sagun, and Riccardo Zecchina. Entropy-sgd: Biasing gradient descent into wide valleys. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124018, 2019.

[15] Rob Brekelmans, Daniel Moyer, Aram Galstyan, and Greg Ver Steeg. Exact rate-distortion in autoencoders via echo noise. In *Advances in Neural Information Processing Systems*, pages 3884–3895, 2019.

[16] Greg Ver Steeg and Aram Galstyan. Maximally informative hierarchical representations of high-dimensional data. In *Artificial Intelligence and Statistics*, pages 1004–1012, 2015.

[17] Alexander A Alemi and Ian Fischer. Therml: Thermodynamics of machine learning. *arXiv preprint arXiv:1807.04162*, 2018.

[18] Stephan Mandt, Matthew D Hoffman, and David M Blei. Stochastic Gradient Descent as Approximate Bayesian Inference. *arXiv:1704.04289*, 2017.

[19] Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. In *Learning in graphical models*, pages 105–161. Springer, 1998.

[20] Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. *arXiv:1312.6114*, 2013.

[21] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 806–813, 2014.

[22] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[23] Carl Doersch and Andrew Zisserman. Multi-task self-supervised visual learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2051–2060, 2017.

[24] Jonathan Baxter. A model of inductive bias learning. *Journal of artificial intelligence research*, 12:149–198, 2000.

[25] Amir R Zamir, Alexander Sax, William Shen, Leonidas J Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3712–3722, 2018.

[26] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

[27] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. *arXiv:1603.05027*, 2016.

[28] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 1520–1528, 2015.

[29] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv:1502.03167*, 2015.

[30] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in neural information processing systems*, pages 2292–2300, 2013.

Appendix: A Free-Energy Principle for Representation Learning

A Related work

We are motivated by the Information Bottleneck (IB) principle of [7, 8], which has been further explored by [9, 10, 11]. The key difference in our work is that while these papers seek to understand the representation for a given task, we focus on how the representation can be adapted to a new task. Further, the Lagrangian (3) has connections to PAC-Bayes bounds [12, 13] and training algorithms that use the free-energy [14]. Our use of rate-distortion for transfer learning is close to the work on unsupervised learning of [15, 16].

This paper builds upon the work of [1, 17]. We refine some results therein, viz., we provide a proof of the convexity of the equilibrium surface and identify it with the equilibrium distribution of SGD. We introduce new ideas such as dynamical processes on the equilibrium surface. Our use of thermodynamics is purely as an inspiration; the work presented here is mathematically rigorous and also provides an immediate algorithmic realization of the ideas.

This paper has strong connections to works that study stochastic processes inspired from statistical physics for machine learning, e.g., approximate Bayesian inference and implicit regularization of SGD [2, 18], variational inference [19, 20]. The iso-classification process instantiates an "automatic" regularization via the trade-off between rate and distortion; this point-of-view is an exciting prospect for future work. The technical content of the paper also draws from optimal transportation [3].

The Lagrangian $F(\lambda, \gamma, \sigma)$ characterizes many different representations of data. The encoder-decoder-classifier architecture is quite generic: formally speaking, an infinite-capacity encoder can characterize any representation of the data. As noted by [17] the Lagrangian $F(\lambda, \gamma, \sigma)$ leads to a number of popular objectives in machine learning, e.g., retaining only C is standard supervised learning; $\lambda = 0$ is the variational objective corresponding to stochastic optimization algorithms [2]; $\lambda = \sigma = 0$ is the variational information bottleneck of [10] to restrict the mutual information between the representation and the data; if one ignores the R term and keep $C + \sigma\, \gamma^{-1} S$ as the objective, one obtains the information bottleneck of [9] which is an alternative version of the original formulation of [7]; $\sigma = \gamma = 0$ gives ELBO or β-VAE [11] as noted before.

A large number of applications begin with pre-trained models [21, 22] or models trained on tasks different [23]. Current methods in transfer learning however do not come with guarantees over the performance on the target dataset, although there is a rich body of older work [24] and ongoing work that studies this [25]. The information-theoretic understanding of transfer and the constrained dynamical processes developed in our paper is a first step towards building such guarantees. In this context, our theory can also be used to tackle catastrophic forgetting [26] to "detune" the model post-training and build up redundant features.

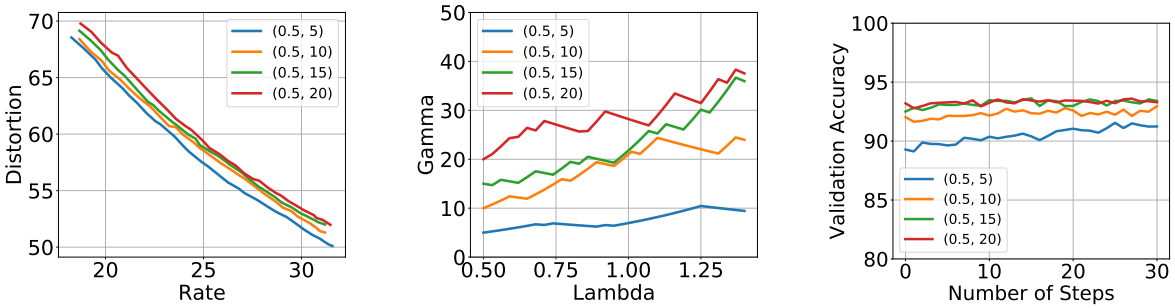
B Details of the experimental setup

Datasets. We use the MNIST [4] and CIFAR-10 [5] datasets for these experiments. The former consists of 28 $\times$28-sized gray-scale images of handwritten digits (60,000 training and 10,000 validation). The latter consists of 32$\times$32-sized RGB images (50,000 training and 10,000 for validation) spread across 10 classes; 4 of these classes (airplane, automobile, ship, truck) are transportation-based while the others are images of

animals and birds.

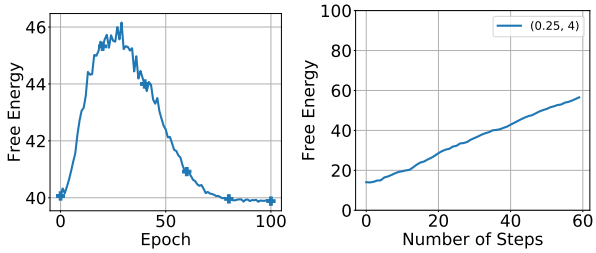
Architecture and training. All models in our experiments consist of an encoder-decoder pair along with a classifier that takes in the latent representation as input. For experiments on MNIST, both encoder and decoder are multi-layer perceptrons with 2 fully-connected layers, the decoder uses a mean-square error loss, i.e., a Gaussian reconstruction likelihood and the classifier consists of a single fully-connected layer. For experiments on CIFAR-10, we use a residual network [27] with 18 layers as an encoder and a decoder with one fully-connected layer and 4 deconvolutional layers [28]. The classifier network for CIFAR-10 is a single fully-connected layer. All models use ReLU non-linearities and batch-normalization [29]. We use Adam [6] to train all models with cosine learning rate annealing.

C More experimental results



(a) (b) (c)

Figure 4: **Iso-classification process for CIFAR-10.** We run 4 different experiments for initial Lagrange multipliers $\lambda = 0.5$ and $\gamma \in \{5, 10, 15, 20\}$. During each experiment, we modify the Lagrange multipliers (Fig. 4b) to keep the classification loss constant and plot the rate-distortion curve (Fig. 4a) along with the validation accuracy (Fig. 4c). The validation loss is constant during each experiment; it takes values between 0.5–0.8 for these initial values of (λ, γ). Similarly, the training loss is constant and takes values between 0.02–0.09 for these initial values of (λ, γ). Observe that the rate-distortion curve in Fig. 4a is much flatter than the one in Fig. 2a which indicates that the model family Θ for CIFAR-10 is much more powerful; this corresponds to the straight line in the RD curve for an infinite model capacity is as shown in Fig. 1.



(a) (b)

Figure 5: **Variation of the free-energy $F(\lambda, \gamma)$ across the equilibration and the iso-classification processes.** Fig. 5a shows the free-energy during equilibration between small changes of (λ, γ). The initial and final values of the Lagrange multipliers are $(0.5, 1)$ and $(0.51, 1.04)$ respectively and the free-energy is about the same for these values. Fig. 5b shows the free-energy as (λ, γ) undergo a large change from their initial value of $(0.25, 4)$ to $(3.5, 26)$ during the iso-classification process in Fig. 2. Since the rate-distortion change a lot (Fig. 2a), the free-energy also changes a lot even if C is constant (Fig. 2c). Number of steps in Fig. 5b refers to the number of steps of running **??**.

D Proof of Lemma 1

The second statement directly follows by observing that F is a minimum of affine functions in (λ, γ). To see the first, evaluate the Hessian of R and F

$$\operatorname{Hess}(R)\operatorname{Hess}(F) = \begin{pmatrix} \frac{\partial^2 R}{\partial D^2} & \frac{\partial^2 R}{\partial D \partial C} \\ \frac{\partial^2 R}{\partial C \partial D} & \frac{\partial^2 R}{\partial C^2} \end{pmatrix} \begin{pmatrix} \frac{\partial^2 F}{\partial \lambda^2} & \frac{\partial^2 F}{\partial \lambda \partial \gamma} \\ \frac{\partial^2 F}{\partial \gamma \partial \lambda} & \frac{\partial^2 F}{\partial \gamma^2} \end{pmatrix}$$

Since we have $F = \min_{e_\theta(z|x), d_\theta(x|z), m_\theta(z)} R + \lambda D + \gamma C$, we obtain

$$\lambda = -\frac{\partial R}{\partial D}, \quad \gamma = -\frac{\partial R}{\partial C}, \quad D = \frac{\partial F}{\partial \lambda}, \quad C = \frac{\partial F}{\partial \gamma}.$$

We then have

$$\begin{aligned} \mathrm{d}\lambda &= -\mathrm{d}\left(\frac{\partial R}{\partial D}\right) \\ &= -\frac{\partial^2 R}{\partial D^2}\,\mathrm{d}D - -\frac{\partial^2 R}{\partial C^2}\,\mathrm{d}C \\ &= -\left(\frac{\partial^2 R}{\partial D^2}\frac{\partial^2 F}{\partial \lambda^2} + \frac{\partial^2 R}{\partial D \partial C}\frac{\partial^2 F}{\partial \lambda \partial \gamma}\right)\mathrm{d}\lambda \\ &\quad - \left(\frac{\partial^2 R}{\partial D^2}\frac{\partial^2 F}{\partial \lambda \partial \gamma} + \frac{\partial^2 R}{\partial D \partial C}\frac{\partial^2 F}{\partial \gamma^2}\right)\mathrm{d}\gamma. \end{aligned}$$

Compare the coefficients on both sides to get

$$\operatorname{Hess}(R)\operatorname{Hess}(F) = -I.$$

Since $0 \succ \operatorname{Hess}(F)$, we have that $\operatorname{Hess}(R) \succ 0$.

E Proof of Lemma 2

We compute the gradient of the objective function as follows.

$$\begin{aligned} \nabla_\theta J(\theta, \lambda, \gamma) &= -\mathop{\mathbb{E}}_{x\sim p(x)} \nabla_\theta \log Z_{\theta,x} \\ &= -\mathop{\mathbb{E}}_{x\sim p(x)} \frac{1}{Z_{\theta,x}} \nabla_\theta Z_{\theta,x} \\ &= -\mathop{\mathbb{E}}_{x\sim p(x)} \frac{1}{Z_{\theta,x}} \int (-\nabla_\theta H)\, \exp(-H)\,\mathrm{d}z \\ &= \mathop{\mathbb{E}}_{x\sim p(x)} \langle \nabla_\theta H \rangle \end{aligned}$$

Then with some effort of computation, we get

$$\begin{aligned} A = \nabla_\theta^2 J(\theta, \lambda, \gamma) &= \nabla_\theta \mathop{\mathbb{E}}_{x\sim p(x)} \Big[\frac{1}{Z_{\theta,x}} \int \nabla_\theta H\, \exp(-H)\,\mathrm{d}z\Big] \\ &= \mathop{\mathbb{E}}_{x\sim p(x)} \left[-\frac{1}{Z_{\theta,x}^2}\left(\int (-\nabla_\theta H)\, \exp(-H)\,\mathrm{d}z\right)\left(\int \nabla_\theta^T H\, \exp(-H)\,\mathrm{d}z\right) + \frac{1}{Z_{\theta,x}} \int \nabla_\theta^2 H\, \exp(-H)\,\mathrm{d}z - \frac{1}{Z_{\theta,x}} \int \nabla_\theta H\, \nabla_\theta^\top H\, \exp \right. \\ &= \mathop{\mathbb{E}}_{x\sim p(x)} \left[\left\langle \nabla_\theta^2 H \right\rangle + \langle \nabla_\theta H \rangle \langle \nabla_\theta H \rangle^\top - \left\langle \nabla_\theta H\, \nabla_\theta^\top H \right\rangle\right]; \end{aligned}$$

$$
\begin{aligned}
b_\lambda &= -\frac{\partial}{\partial\lambda}\nabla_\theta J = \frac{\partial}{\partial\lambda}\mathop{\mathbb{E}}_{x\sim p(x)}\left[\frac{1}{Z_{\theta,x}}\int \nabla_\theta H\,\exp(-H)\,\mathrm{d}z\right] \\
&= \mathop{\mathbb{E}}_{x\sim p(x)}\left[-\frac{1}{Z_{\theta,x}^2}\left(\int -\frac{\partial H}{\partial\lambda}\,\exp(-H)\,\mathrm{d}z\right)\left(\int \nabla_\theta H\,\exp(-H)\,\mathrm{d}z\right)+\frac{1}{Z_{\theta,x}}\int \frac{\partial}{\partial\lambda}\nabla_\theta H\,\exp(-H)\,\mathrm{d}z-\frac{1}{Z_{\theta,x}}\int \frac{\partial H}{\partial\lambda}\nabla_\theta H\,\exp(-H)\,\mathrm{d}z\right] \\
&= -\mathop{\mathbb{E}}_{x\sim p(x)}\left[\left\langle\frac{\partial\nabla_\theta H}{\partial\lambda}\right\rangle-\left\langle\frac{\partial H}{\partial\lambda}\,\nabla_\theta H\right\rangle+\left\langle\frac{\partial H}{\partial\lambda}\right\rangle\langle\nabla_\theta H\rangle\right]; \\
b_\gamma &= -\frac{\partial}{\partial\gamma}\nabla_\theta J = \frac{\partial}{\partial\gamma}\mathop{\mathbb{E}}_{x\sim p(x)}\left[\frac{1}{Z_{\theta,x}}\int \nabla_\theta H\,\exp(-H)\,\mathrm{d}z\right] \\
&= \mathop{\mathbb{E}}_{x\sim p(x)}\left[-\frac{1}{Z_{\theta,x}^2}\left(\int -\frac{\partial H}{\partial\gamma}\,\exp(-H)\,\mathrm{d}z\right)\left(\int \nabla_\theta H\,\exp(-H)\,\mathrm{d}z\right)+\frac{1}{Z_{\theta,x}}\int \frac{\partial}{\partial\gamma}\nabla_\theta H\,\exp(-H)\,\mathrm{d}z-\frac{1}{Z_{\theta,x}}\int \frac{\partial H}{\partial\gamma}\,\nabla_\theta H\,\exp(-H)\,\mathrm{d}z\right] \\
&= -\mathop{\mathbb{E}}_{x\sim p(x)}\left[\left\langle\frac{\partial\nabla_\theta H}{\partial\gamma}\right\rangle-\left\langle\frac{\partial H}{\partial\gamma}\,\nabla_\theta H\right\rangle+\left\langle\frac{\partial H}{\partial\gamma}\right\rangle\langle\nabla_\theta H\rangle\right].
\end{aligned}
$$

According to the quasi-static constraints (6), we have:

$$
A\dot\theta + \dot\lambda b_\lambda + \dot\gamma b_\gamma = 0,
$$

which implies

$$
\begin{aligned}
\dot\theta &= A^{-1}b_\lambda\,\dot\lambda + A^{-1}b_\gamma\,\dot\gamma \\
&= \theta_\lambda\dot\lambda + \theta_\gamma\dot\gamma
\end{aligned}
$$

F Computation of Iso-classification constraint

We start with computing the gradient of classification loss, clear that $C = \mathbb{E}_{x\sim p(x)}\left[-\int \mathrm{d}z\; e(z|x)\log c(y|z)\right] = -\,\mathbb{E}_{x\sim p(x)}\langle\ell\rangle$, where $\ell = \log c_\theta(y_x|z)$ is the logarithm of the classification loss, then

$$
\begin{aligned}
\nabla_\theta C &= -\nabla_\theta \mathop{\mathbb{E}}_{x\sim p(x)}\left[\frac{1}{Z_{\theta,x}}\int \ell\,\exp(-H)\,\mathrm{d}z\right] \\
&= -\mathop{\mathbb{E}}_{x\sim p(x)}\left[-\frac{1}{Z_{\theta,x}^2}\left(\int(-\nabla_\theta H)\,\exp(-H)\,\mathrm{d}z\right)\left(\int \ell\,\exp(-H)\,\mathrm{d}z\right)+\frac{1}{Z_{\theta,x}}\int \nabla_\theta\,\ell\,\exp(-H)\,\mathrm{d}z-\frac{1}{Z_{\theta,x}}\int \ell\,\nabla_\theta H\,\exp(-H)\,\mathrm{d}z\right] \\
&= -\mathop{\mathbb{E}}_{x\sim p(x)}\left[\langle\nabla_\theta\,\ell\rangle+\langle\nabla_\theta H\rangle\,\langle\ell\rangle-\langle\ell\,\nabla_\theta H\rangle\right];
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial}{\partial\lambda}C &= -\frac{\partial}{\partial\lambda}\mathop{\mathbb{E}}_{x\sim p(x)}\left[\frac{1}{Z_{\theta,x}}\int \ell\,\exp(-H)\,\mathrm{d}z\right] \\
&= -\mathop{\mathbb{E}}_{x\sim p(x)}\left[-\frac{1}{Z_{\theta,x}^2}\left(\int -\frac{\partial H}{\partial\lambda}\,\exp(-H)\,\mathrm{d}z\right)\left(\int \ell\,\exp(-H)\,\mathrm{d}z\right)-\frac{1}{Z_{\theta,x}}\int \ell\,\frac{\partial H}{\partial\lambda}\,\exp(-H)\,\mathrm{d}z\right] \\
&= -\mathop{\mathbb{E}}_{x\sim p(x)}\left[\left\langle\frac{\partial H}{\partial\lambda}\right\rangle\langle\ell\rangle-\left\langle\ell\,\frac{\partial H}{\partial\lambda}\right\rangle\right];
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial}{\partial\gamma}C &= -\frac{\partial}{\partial\gamma}\mathop{\mathbb{E}}_{x\sim p(x)}\left[\frac{1}{Z_{\theta,x}}\int \ell\,\exp(-H)\,\mathrm{d}z\right] \\
&= -\mathop{\mathbb{E}}_{x\sim p(x)}\left[-\frac{1}{Z_{\theta,x}^2}\left(\int -\frac{\partial H}{\partial\lambda}\,\exp(-H)\,\mathrm{d}z\right)\left(\int \ell\,\exp(-H)\,\mathrm{d}z\right)-\frac{1}{Z_{\theta,x}}\int \ell\,\frac{\partial H}{\partial\gamma}\,\exp(-H)\,\mathrm{d}z\right] \\
&= -\mathop{\mathbb{E}}_{x\sim p(x)}\left[\left\langle\frac{\partial H}{\partial\gamma}\right\rangle\langle\ell\rangle-\left\langle\ell\,\frac{\partial H}{\partial\gamma}\right\rangle\right].
\end{aligned}
$$

The iso-classification loss constrains together with quasi-static constrains imply that:

$$
\begin{aligned}
0 &= \dot{\theta}^\top \nabla_\theta C + \dot{\lambda}\frac{\partial C}{\partial \lambda} + \dot{\gamma}\frac{\partial C}{\partial \gamma} \\
&= \dot{\lambda}\left(\theta_\lambda^\top \nabla_\theta C + \frac{\partial C}{\partial \lambda}\right) + \dot{\gamma}\left(\theta_\gamma^\top \nabla_\theta C + \frac{\partial C}{\partial \gamma}\right) \\
&= -\dot{\lambda} \mathop{\mathbb{E}}_{x\sim p(x)}\left[\left\langle \frac{\partial H}{\partial \lambda}\right\rangle \langle \ell \rangle - \left\langle \ell \, \frac{\partial H}{\partial \lambda}\right\rangle + \left\langle \theta_\lambda^\top \nabla_\theta H\right\rangle \langle \ell \rangle - \left\langle \ell \theta_\lambda^\top \nabla_\theta H\right\rangle + \left\langle \theta_\lambda^\top \nabla_\theta \ell\right\rangle\right] \\
&\qquad -\dot{\gamma} \mathop{\mathbb{E}}_{x\sim p(x)}\left[\left\langle \frac{\partial H}{\partial \gamma}\right\rangle \langle \ell \rangle - \left\langle \ell \, \frac{\partial H}{\partial \gamma}\right\rangle + \left\langle \theta_\gamma^\top \nabla_\theta H\right\rangle \langle \ell \rangle - \left\langle \ell \theta_\gamma^\top \nabla_\theta H\right\rangle + \left\langle \theta_\gamma^\top \nabla_\theta \ell\right\rangle\right] \\
&= C_\lambda \dot{\lambda} + C_\gamma \dot{\gamma},
\end{aligned}
$$

where the second equation is followed by the quasi-static constrains.

G Iso-classification equations for changing data distribution

In this section we analyze the dynamics for iso-classification loss process when the data distribution evolves with time. $\frac{\partial p(x)}{\partial t}$ will lead to additional terms that represent the partial derivatives with respect to t on both the quasi-static and iso-classification constrains. More precisely, have:

$$
\begin{aligned}
b_t = \frac{\partial}{\partial t}\nabla_\theta J = \int \frac{\partial p(x)}{\partial t} \langle \nabla_\theta H \rangle \, ; \\
\frac{\partial}{\partial t} C = -\int \frac{\partial p(x)}{\partial t} \langle \ell \rangle \, ,
\end{aligned}
$$

next the quasi-static and iso-classification constraints are ready to be modified as

$$
\begin{aligned}
0 \equiv \frac{\mathrm{d}}{\mathrm{d}t}\nabla_\theta J(\theta,\lambda,\gamma) &\iff 0 = \nabla_\theta^2 F \, \dot{\theta} + \dot{\lambda} \, \frac{\partial \nabla_\theta F}{\partial \lambda} + \dot{\gamma} \, \frac{\partial \nabla_\theta F}{\partial \gamma} + \frac{\partial \nabla_\theta F}{\partial t} \\
&\iff \dot{\theta} = -\dot{\lambda} \, A^{-1} \, b_\lambda - \dot{\gamma} \, A^{-1} \, b_\gamma - A^{-1} \, b_t \\
&\iff \dot{\theta} = \dot{\lambda} \, \theta_\lambda + \dot{\gamma} \, \theta_\gamma + \theta_t
\end{aligned} \quad ;
$$

$$
\begin{aligned}
0 \equiv \frac{\mathrm{d}}{\mathrm{d}t} C &\iff 0 = \dot{\theta}^\top \nabla_\theta C + \dot{\lambda}\frac{\partial C}{\partial \lambda} + \dot{\gamma}\frac{\partial C}{\partial \gamma} + \frac{\partial C}{\partial t} \\
&\iff 0 = \dot{\lambda}\left(\theta_\lambda^\top \nabla_\theta C + \frac{\partial C}{\partial \lambda}\right) + \dot{\gamma}\left(\theta_\gamma^\top \nabla_\theta C + \frac{\partial C}{\partial \gamma}\right) + \left(\theta_t + \frac{\partial C}{\partial t}\right) \\
&\iff 0 = \dot{\lambda} \, C_\lambda + \dot{\gamma} \, C_\gamma + C_t,
\end{aligned}
$$

H Geodesic transfer

The dynamics of Lemma 3 is valid for any $(\dot{\lambda}, \dot{\gamma})$. We provide a locally optimal way to change (λ, γ) in this section. Note that

$$
\begin{aligned}
\dot{C} &= 0, \\
\dot{D} &= \frac{\partial D}{\partial \lambda}\dot{\lambda} + \frac{\partial D}{\partial \gamma}\dot{\gamma} = -\alpha\left(\frac{\partial^2 F}{\partial \lambda^2}\frac{\partial^2 F}{\partial \gamma^2} - \left(\frac{\partial^2 F}{\partial \lambda \partial \gamma}\right)^2\right) \\
&= -\alpha \det\left(\mathrm{Hess}(F)\right), \\
\dot{R} &= \frac{\partial R}{\partial D}\dot{D} + \frac{\partial R}{\partial C}\dot{C} = -\lambda \dot{D}.
\end{aligned} \tag{11}
$$

The first equality is simply our iso-classification constraint. For $\alpha > 0$, the second one indicates that $\dot{D} < 0$ using Lemma 1 which shows that $0 \succ \text{Hess}(F)$. This also gives $\dot{\lambda} > 0$ in (8). The third equality is a powerful observation: it indicates a trade-off between rate and distortion, if $\dot{D} < 0$ we have $\dot{R} > 0$. It also shows the geometric structure of the equilibrium surface by connecting $\dot{R}$ and $\dot{D}$ together, which we will exploit next.

Computing the functionals R, D and C during the iso-classification transfer presents us with a curve in RDC space. Geodesic transfer implies that the functionals R, D follow the shortest path in this space. But notice that if **we assume that the model capacity is infinite**, the RDC space is Euclidean and therefore the geodesic is simply a straight line. Since we keep the classification loss constant during the transfer, $\dot{C} = 0$, straight line implies that slope $\mathrm{d}D/\mathrm{d}R$ is a constant, say k. Thus $\dot{D} = k\dot{R}$. Observe that $\dot{R} = \frac{\partial R}{\partial D}\dot{D} + \frac{\partial R}{\partial C}\dot{C} + \frac{\partial R}{\partial t} = -\lambda\dot{D} + \frac{\partial R}{\partial t}$. Combining the iso-classification constraint and the fact that $\dot{D} = k\dot{R} = -k\lambda\dot{D} + k\frac{\partial R}{\partial t}$, gives us a linear system:

$$\begin{aligned} \frac{\partial D}{\partial \lambda}\dot{\lambda} + \frac{\partial D}{\partial \gamma}\dot{\gamma} &= \frac{k\frac{\partial R}{\partial t}}{1+k\lambda}; \\ \frac{\partial C}{\partial \lambda}\dot{\lambda} + \frac{\partial C}{\partial \gamma}\dot{\gamma} + \frac{\partial C}{\partial t} &= 0 \end{aligned} \tag{12}$$

Equivalently, we have

$$\text{Hess}(F) \begin{pmatrix} \dot{\lambda} \\ \dot{\gamma} \end{pmatrix} = \begin{pmatrix} -\frac{k\frac{\partial R}{\partial t}}{1+k\lambda} \\ -\frac{\partial C}{\partial t} \end{pmatrix}$$

We solve this system to update (λ, γ) during the transfer.

I Optimally transporting the data distribution

We first give a brief description of the theory of optimal transportation. The optimal transport map between the source task and the target task will be used to define a dynamical process for the task. We only compute the transport for the inputs x between the source and target distributions and use a heuristic to obtain the transport for the labels y. This choice is made only to simplify the exposition; it is straightforward to handle the case of transport on the joint distribution $p(x, y)$.

If i.i.d samples from the source task are denoted by $\{x_1^s, \ldots, x_{n_s}^s\}$ and those of the target distribution are $\{x_1^t, \ldots, x_{n_t}^t\}$ the empirical source and target distributions can be written as

$$p^s(x) = \frac{1}{n_s}\sum_{i=1}^{n_s} \delta_{x-x_i^s}, \text{ and } p^t(x) = \frac{1}{n_t}\sum_{i=1}^{n_t} \delta_{x-x_i^t}$$

respectively; here $\delta_{x-x'}$ is a Dirac delta distribution at x'. Since the empirical data distribution is a sum of a finite number of Dirac measures, this is a discrete optimal transport problem and easy to solve. We can use the Kantorovich relaxation to denote by $\mathcal{B}$ the set of probabilistic couplings between the two distributions:

$$\mathcal{B} = \left\{\Gamma \in \mathbb{R}_+^{n_s \times n_t} : \Gamma \mathbf{1}_{n_s} = p, \Gamma^\top \mathbf{1}_{n_s} = q\right\}$$

where $\mathbf{1}_n$ is an n-dimensional vector of ones. The Kantorovich formulation solves for

$$\Gamma^* = \underset{\Gamma \in \mathcal{B}}{\text{argmin}} \sum_{i=1}^{n_s}\sum_{t=1}^{n_t} \Gamma_{ij}\, \kappa_{ij} \tag{13}$$

where $\kappa \in \mathbb{R}_+^{n_s \times n_t}$ is a cost function that models transporting the datum x_i^s to x_j^t. This is the metric of the underlying data domain and one may choose any reasonable metric for $\kappa = \|x_i^s - x_j^t\|_2^2$. The problem in (13) is a convex optimization problem and can be solved easily; in practice we use the Sinkhorn's algorithm [30] which adds an entropic regularizer $-h(\Gamma) = \sum_{ij} \Gamma_{ij} \log \Gamma_{ij}$ to the objective in (13).

I.1 Changing the data distribution

Given the optimal probabilistic coupling Γ^* between the source and the target data distributions, we can interpolate between them at any $t \in [0,1]$ by following the geodesics of the Wasserstein metric

$$p(x,t) = \operatorname*{argmin}_p \ (1-t)W_2^2(p^s,p) + tW_2^2(p,p^t).$$

For discrete optimal transport problems, as shown in [3], the interpolated distribution p_t for the metric $\kappa_{ij} = \|x_i^2 - x_j^t\|_2^2$ is given by

$$p(x,t) = \sum_{i=1}^{n_s}\sum_{j=1}^{n_t} \Gamma_{ij}^* \ \delta_{x-(1-t)x_i^s - tx_j^t}. \tag{14}$$

Observe that the interpolated data distribution equals the source and target distribution at $t=0$ and $t=1$ respectively and it consists of linear interpolations of the data in between.

Remark 4 (Interpolating the labels). The interpolation in (14) gives the marginal on the input space interpolated between the source and target tasks. To evaluate the functionals in Section 3 for the classification setting, we would also like to interpolate the labels. We do so by setting the true label of the interpolated datum $x = (1-t)x_i^s + tx_j^t$ to be linear interpolation between the source label and the target label.

$$y(x,t) = (1-t)\delta_{y-y_{x_i^s}} + t\delta_{y-y_{x_j^t}}$$

for all i,j. Notice that the interpolated distribution $p(x,t)$ is a sum of Dirac delta distributions weighted by the optimal coupling. We therefore only need to evaluate the labels at all the interpolated data.

Remark 5 (Linear interpolation of data). Our formulation of optimal transportation leads to a linear interpolation of the data in (14). This may not work well for image-based data where the square metric $\kappa_{ij} = \|x_i^s - x - k^t\|_2^2$ may not be the appropriate metric. We note that this interpolation of data is an artifact of our choice of κ_{ij}, other choices for the metric also fit into the formulation and should be viable alternatives if they result in efficient computation.

J Transfer learning between two subsets of CIFAR-10

The iso-classification process is a quasi-static process, i.e., the model parameters θ are lie on the equilibrium surface at all times $t \in [0,1]$ during the transfer. Note that both the equilibrium surface and the free-energy $F(\lambda,\gamma)$ are functions of the data and change with time. Let us write this explicitly as

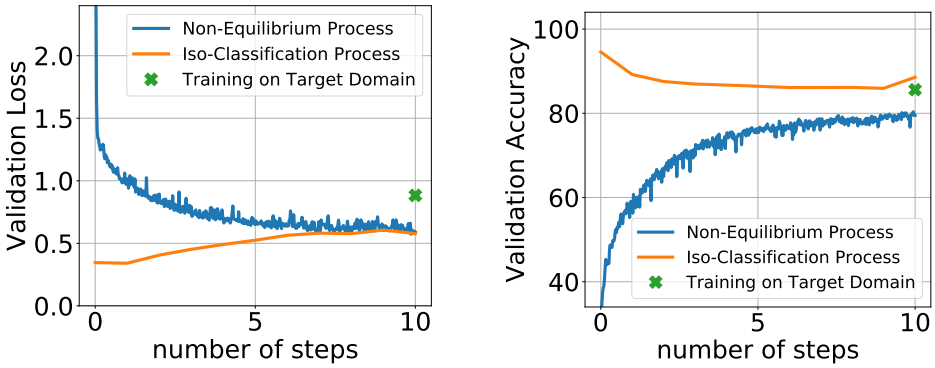
$$F(t) := R(t,\lambda(t),\gamma(t)) + \lambda D(t,\lambda(t),\gamma(t)) + \gamma C_0$$

where C_0 is the classification loss. We prescribed a geodesic transfer above where the Lagrange multipliers λ,γ were adapted simultaneously to confirm to the constraints of the equilibrium surface locally. We can forgot this and instead adapt them using the following heuristic. We let $\dot\lambda = k$ for some constant k and use

$$\frac{\partial C}{\partial \lambda}\dot\lambda + \frac{\partial C}{\partial \gamma}\dot\gamma + \frac{\partial C}{\partial t} = 0, \tag{15}$$

to get the evolution curve of $\gamma(t)$.

Here we present experimental results of an iso-classification process for transferring the learnt representation. We pick the source dataset to be all "vehicles" (airplane, automobile, ship and truck) in CIFAR-10 and the target dataset consists of four "animals" (bird, cat, deer and dog). We let the output size of classifier be



(a) (b)

Figure 6: **Transferring from source dataset of CIFAR vehicles to the target dataset consisting of four animals.** Fig. 6a shows the variation of validation loss during the transfer. Fig. 6b shows the validation accuracy during the transfer. The orange curve corresponds to iso-classification transfer; the blue curve is the result of directly fine-tuning the source model on the target data (note the very low accuracy at the start); the green point is the accuracy of training on the target task from scratch.

four. Except the output size of classifier, we use the exactly same model that searching for iso-classification process on the equilibrium surface of CIFAR-10. Our goal is to adapt a model trained on the source task to the target task while keeping its classification loss constant. We run the iso-classification transfer dynamics (15) and the results are shown in Fig. 6.

It is evident that both the classification accuracy and loss are constants throughout the transfer. CIFAR-10 is a more complex dataset as comparing with MNIST and the accuracy gap between iso-classification transfer, fine-tuning from the source and training from scratch is significant. Observe that the classification loss gap between iso-classification transfer and training from scratch on the target is also significant. The benefit of running the iso-classification transfer is that we can be guaranteed about the final accuracy and validation loss of the model.

J.1 Details of the experimental setup for CIFAR-10

At moment t, parameters λ, γ determine our objective free-energy. We compute iso-classification loss transfer process by first setting initial states: $(\lambda = 4, \gamma = 100)$. We train on source dataset for 300 epochs with Adam and a learning rate of 10^{-3} that drops by a factor of 10 after every 120 epochs to obtain the initial state. Every moment we let λ, γ and t change a little bit and then apply the beta function learning rate schedule to achieve the transition between equilibrium states. We compute the partial derivatives $\frac{\partial C}{\partial t}$, $\frac{\partial C}{\partial \lambda}$ and $\frac{\partial C}{\partial \gamma}$ by using finite difference. For each moment t we transferring, equipping with these partial derivatives and solving the equation (15) leads to solution for $\dot{\gamma}$, where $\dot{\lambda}$ is a constant, in our experiment we set $\dot{\lambda} = -1.5$. $\dot{\gamma}$ will enable us to adjust the updates for γ.