A. Details of the experimental setup

A.1. Architecture and training.

We show results using an 8-layer convolutional neural network with ReLU nonlinearities, dropout, batch-normalization with a final fully-connected layer. The larger model used for experiments in Fig. 5 is a wide-residual-network (WRN-16-4 architecture of (Zagoruyko & Komodakis, 2016)).

A.2. Transferring between CIFAR-10 and CIFAR-100

We consider four tasks: (i) all vehicles (airplane, automobile, ship, truck) in CIFAR-10, consisting of 20,000 32×32-sized RGB images; (ii) the remainder, namely six animals in CIFAR-10, consisting of 30,000 32×32-sized RGB images; (iii) the entire CIFAR-10 dataset and (iv) the entire CIFAR-100 dataset, consisting of 50,000 images and spread across 100 classes.

We pre-train model on source tasks using stochastic gradient descent (SGD) for 60 epochs, with mini-batch size of 20, learning rate schedule is set to 10^{-3} for epochs 0 – 40 and 8×10^{-4} for epochs 40 – 60. When CIFAR-100 is the source dataset, we train for 180 epochs with the learning rate set to 10^{-3} for epochs 0 – 120, and 8×10^{-4} for epochs 120 – 180.

We chose a slightly smaller version of the source and target datasets to compute the distance, each of them have 19,200 images. The class distribution on all source and target classes is balanced. We did this to reduce the size of the coupling matrix Γ in (12a). The coupling matrix connecting inputs in the source and target datasets is $\Gamma \in \mathbb{R}^{19200\times 19200}$ which is still quite large to be tractable during optimization. We therefore use a block diagonal approximation of the coupling matrix; 640 blocks are constructed each of size 30×30 and all other entries in the coupling matrix are set to zero at the beginning of each iteration in (12a) after computing the dense coupling matrix using the linear program. This effectively entails that the set of couplings over which we compute the transport is not the full convex polytope in Sec. 2.2 but rather a subset of it. We sample a mini-batch of 20 images from the interpolated distribution corresponding to this block-diagonal coupling matrix for each weight update of (12c). We run 40 epochs, i.e., with 19200/20 = 960 weight updates per epoch for computing the weight trajectory at *each iteration* k in (12). The learning rate is fixed to 8×10^{-4} in the transfer learning phase.

A.3. Transferring among subsets of CIFAR-100

The same 8-layer convolutional network is used to show results for transfer between subsets of CIFAR-10 and CIFAR-100. CIFAR-10 is split into the two tasks animals and vehicle again. We construct five tasks (herbivores, carnivores, vehicles-1,vehicles-2 and flowers) that are subsets of the CIFAR-100 dataset. Each of these tasks consists of 5 sub-classes.

We train the model on the source task using SGD for 400 epochs with a mini-batch size of 20. Learning rate is set to 10^{-3} for epochs 0 – 240, and to 8×10^{-4} for epochs 240 – 400.

Tasks that are subsets of CIFAR-100 in the experiments in this section have few samples (2500 each) so we select 2400 images from source and target datasets respectively; we could have chosen a larger source dataset when transferring from CIFAR-10 animals or vehicles but we did not so for sake of simplicity. The number 2400 was chosen to make the block diagonal approximation of the coupling matrix have 120×120 entries in each block; this was constrained by the GPU memory. The coupling matrix Γ therefore has 2400×2400 entries with 20 blocks on the diagonal.

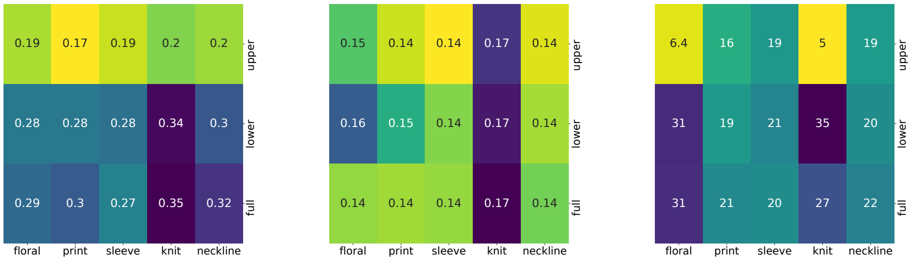
Again, we use a mini-batch size of 20 for 240 epochs (2400/20 = 120 weight updates per epoch) during the transfer from the source dataset to the target dataset. The learning rate is fixed to 8×10^{-4} in the transfer learning phase.

A.4. Training setup for wide residual network

We pre-train WRN-16-4 on source tasks using SGD for 400 epochs with a mini-batch size of 20. Learning rate is 10^{-1} for epochs 0 – 120, 2×10^{-2} for epochs 120 – 240, 4×10^{-3} for epochs 240–320, and 8×10^{-4} for epochs 320 – 400. Other experimental details are the same as those in Sec. A.3.

B. Experiments on the Deep Fashion dataset

For the Deep Fashion dataset (Liu et al., 2016), we consider three binary category classification tasks (upper clothes, lower clothes, and full clothes) and five binary attribute classification tasks (floral, print, sleeve, knit, and neckline). We show results in Fig. 8 using 3× 5 distance matrices where numbers in each cell indicate the distance between the source task



(a) **(b)** **(c)**

Figure 8. Fig. 8a shows distances (numbers in the cell) among sub-tasks in DeepFashion computed using our coupled transfer process (r = 0.37, p = 0.33), Fig. 8c shows distances estimated using Task2Vec (r = 0.04, p = 0.75) while Fig. 8c shows distances estimated using fine-tuning (r = 0.54, p = 0.36 with itself). Numerical values of the distances in this figure are not comparable with each other. Coupled transfer, Task2Vec and fine-tuning all agree with that transferring to knit is relatively hard. Transferring from upper-cloth to knit is easy via fine-tuning and coupled transfer correctly estimates this distance to be small; the distance estimated by Task2Vec is much larger in comparison. Since these matrices are non-square, we ran the Mantel test for three 3×3 submatrices (sweep across columns) of these 3×5 matrices and report the mean test statistic and the average p-value across these tests above.

(row) and the target task (column). We show results using a wide-residual-network (WRN-16-4, (Zagoruyko & Komodakis, 2016)).

The model is trained using SGD for 400 epochs with a mini-batch size 20. Learning rate is 10^{-1} for epochs 0 – 120, 2×10^{-2} for epochs 120 – 240, 4×10^{-3} for epochs 240–320, and 8×10^{-4} for epochs 320 – 400. We sample 14,000 images from the source and target datasets to compute distances. A mini-batch size of 20 is used during transfer and we run (12c) for 60 epochs (14000/20 = 700 weight updates per epoch).

C. Proof of Thm. 5

We first prove a simpler theorem.

Theorem 6. Given a trajectory of the weights $\{w(\tau)\}_{\tau\in[0,1]}$ and a sequence $0 \leq \tau_1 < \tau_2 < ... < \tau_K \leq 1$, then for all $\epsilon > \frac{2}{K}\sum_{k=1}^{K} \mathcal{R}_N(\|w(\tau_k)\|_{\text{FR}})$, the probability that

$$\frac{1}{K}\sum_{k=1}^{K}\left(\mathop{\mathbb{E}}_{(x,y)\sim p_{\tau_k}}[\ell(\omega(\tau_k),x,y)] - \frac{1}{N}\sum_{(x,y)\sim \hat{p}_{\tau_k}} \ell(\omega(\tau_k),x,y)\right)$$

is greater than ϵ is upper bounded by

$$\exp\left\{-\frac{2K}{M^2}\left(\epsilon - \frac{2}{K}\sum_{k=1}^{K}\mathcal{R}_N(\|w(\tau_k)\|_{\text{FR}})\right)^2\right\}. \tag{15}$$

Proof. For each moment τ_k, by taking supremum

$$\mathop{\mathbb{E}}_{(x,y)\sim p_{\tau_k}} \ell(w(\tau_k),x,y) - \frac{1}{N}\sum_{(x,y)\sim\hat{p}_{\tau_k}} \ell(w(\tau_k),x,y) \leq \sup_{\|w\|_{\text{FR}}\leq\|w(\tau_k)\|_{\text{FR}}}\left(\mathop{\mathbb{E}}_{(x,y)\sim p_{\tau_k}} \ell(w,x,y) - \frac{1}{N}\sum_{(x,y)\sim\hat{p}_{\tau_k}} \ell(w,x,y)\right), \tag{16}$$

where $\|\cdot\|_{\text{FR}}$ denotes Fisher-Rao norm (Liang et al., 2019). The right hand side of inequality(16) is a random variable that depends on the drawn sampling set $\hat{p}_{\tau_k}$ with size N. Denoting

$$\phi(\hat{p}_{\tau_k}) := \sup_{\|w\|_{\text{FR}}\leq\|w(\tau_k)\|_{\text{FR}}}\left(\mathop{\mathbb{E}}_{(x,y)\sim p_{\tau_k}} \ell(w,x,y) - \frac{1}{N}\sum_{(x,y)\sim\hat{p}_{\tau_k}} \ell(w,x,y)\right), \tag{17}$$

We would like to bound the expectation of $\phi(\hat{p}_{\tau_k})$ in terms of the Rademacher complexity. In order to do this, we introduce a "ghost sample" with size N, $\hat{p}'_{\tau_k}$, independently drawn identically from $p_{\tau_k}(x,y)$, we rewrite the expectations

$$\begin{aligned}
\mathop{\mathbb{E}}_{\hat p_{\tau_k}} \phi(\hat p_{\tau_k}) &= \mathop{\mathbb{E}}_{\hat p_{\tau_k}} \left[\sup_{\|w\|_{\text{FR}} \le \|w(\tau_k)\|_{\text{FR}}} \left(\mathop{\mathbb{E}}_{(x,y)\sim p_{\tau_k}} \ell(w,x,y) - \frac{1}{N} \sum_{(x,y)\sim \hat p_{\tau_k}} \ell(w,x,y) \right) \right] \\
&= \mathop{\mathbb{E}}_{\hat p_{\tau_k}} \left[\sup_{\|w\|_{\text{FR}} \le \|w(\tau_k)\|_{\text{FR}}} \mathop{\mathbb{E}}_{\hat p'_{\tau_k}} \left(\frac{1}{N} \sum_{(x,y)\sim \hat p'_{\tau_k}} \ell(w,x,y) - \frac{1}{N} \sum_{(x,y)\sim \hat p_{\tau_k}} \ell(w,x,y) \right) \right] \\
&\le \mathop{\mathbb{E}}_{\hat p_{\tau_k}, \hat p'_{\tau_k}, \sigma} \left[\sup_{\|w\|_{\text{FR}} \le \|w(\tau_k)\|_{\text{FR}}} \frac{1}{N} \left(\sum_{(x,y)\sim \hat p_{\tau_k}} \sigma^i (\ell(w,x,y) - \ell(w,x,y)) \right) \right] \\
&\le \mathop{\mathbb{E}}_{\hat p_{\tau_k}, \sigma} \left[\sup_{\|w\|_{\text{FR}} \le \|w(\tau_k)\|_{\text{FR}}} \frac{1}{N} \sum_{(x,y)\sim \hat p_{\tau_k}} \sigma^i \ell(w,x,y) \right] + \mathop{\mathbb{E}}_{\hat p_{\tau_k}, \sigma} \left[\sup_{\|w\|_{\text{FR}} \le \|w(\tau_k)\|_{\text{FR}}} \frac{1}{N} \sum_{(x,y)\sim \hat p_{\tau_k}} \sigma^i \ell(w,x,y) \right] \\
&= 2\mathcal{R}_N(\|w(\tau_k)\|_{\text{FR}}),
\end{aligned}$$

where $\sigma = (\sigma^1, \sigma^2, \dots, \sigma^N)$ are independent random variables drawn from the Rademacher distribution, the last equality is followed by the definition of Rademacher Complexity within $\|w(\tau_k)\|_{\text{FR}}$-ball in the Fisher-Rao norm. By Hoeffding's lemma, for $\lambda > 0$

$$\begin{aligned}
\mathop{\mathbb{E}}_{\hat p_{\tau_k}} \exp\left\{ \lambda \left(\mathop{\mathbb{E}}_{(x,y)\sim p_{\tau_k}} \ell(w(\tau_k),x,y) - \frac{1}{N} \sum_{(x,y)\sim \hat p_{\tau_k}} \ell(w(\tau_k),x,y) \right) \right\} &= \mathop{\mathbb{E}}_{\hat p_{\tau_k}} e^{\lambda \phi(\hat p_{\tau_k})} \\
&\le e^{\lambda \mathbb{E}_{\hat p_{\tau_k}} \phi(\hat p_{\tau_k}) + \frac{\lambda^2 M^2}{8}} \\
&\le e^{2\lambda \mathcal{R}_N(\|w(\tau_k)\|_{\text{FR}}) + \frac{\lambda^2 M^2}{8}}.
\end{aligned} \tag{18}$$

For each moment τ_k, we have inequality(18), which implies

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\hat p_{\tau_k}:\ 1\le k\le K} \exp\left\{ \lambda \sum_{k=1}^{K} \left(\mathop{\mathbb{E}}_{(x,y)\sim p_{\tau_k}} \ell(w(\tau_k),x,y) - \frac{1}{N} \sum_{(x,y)\sim \hat p_{\tau_k}} \ell(w(\tau_k),x,y) \right) \right\} \\
&= \prod_{k=1}^{K} \mathop{\mathbb{E}}_{\hat p_{\tau_k}} \exp\left\{ \lambda \left(\mathop{\mathbb{E}}_{(x,y)\sim p_{\tau_k}} \ell(w(\tau_k),x,y) - \frac{1}{N} \sum_{(x,y)\sim \hat p_{\tau_k}} \ell(w(\tau_k),x,y) \right) \right\} \\
&\le \exp\left\{ \sum_{k=1}^{K} \left[2\lambda \mathcal{R}_N(\|w(\tau_k)\|_{\text{FR}}) + \frac{\lambda^2 M^2}{8} \right] \right\}.
\end{aligned}$$

Finally for all $K\epsilon > 2\sum_{k=1}^{K} \mathcal{R}_N(\|w(\tau_k)\|_{\text{FR}})$, by Markov's inequality

$$\begin{aligned}
&Pr\left\{ \sum_{k=1}^{K} \left(\mathop{\mathbb{E}}_{(x,y)\sim p_{\tau_k}} \ell(w(\tau_k),x,y) - \frac{1}{N} \sum_{(x,y)\sim \hat p_{\tau_k}} \ell(w(\tau_k),x,y) \right) > K\epsilon \right\} \\
&\le \exp\left\{ -\lambda K\epsilon + \sum_{k=1}^{K} \left[2\lambda \mathcal{R}_N(\|w(\tau_k)\|_{\text{FR}}) + \frac{\lambda^2 M^2}{8} \right] \right\}
\end{aligned} \tag{19}$$

Put $\lambda = \frac{4K\left(\epsilon - \frac{2}{K}\sum_{k=1}^{K} \mathcal{R}_N(\|w(\tau_k)\|_{\text{FR}})\right)}{M^2}$ in right hand side of inequality(19), then we finish the proof. □

Proof of Thm. 5

The upper bound in (19) above states that we should minimize the Rademacher complexity of the hypothesis space in order to ensure that the weight trajectory has a small generalization gap at all time instants. For linear models, as discussed in the main paper (Liang et al., 2019), the Rademacher complexity can be related to the Fisher-Rao norm $\langle w, gw \rangle$. The Fisher-Rao distance on the manifold, namely

$$\int_0^1 \mathop{\mathbb{E}}_{x\sim p_\tau(x)} \left[\sqrt{2\mathrm{KL}\left(p_{w(\tau)}(\cdot|x),\ p_{w(\tau+\mathrm{d}\tau)}(\cdot|x)\right)}\right] \mathrm{d}\tau = \int_0^1 \mathop{\mathbb{E}}_{x\sim p_\tau(x)} \sqrt{\left\langle \dot{w(\tau)}, g(w(\tau))\dot{w(\tau)} \right\rangle}\, \mathrm{d}\tau \tag{20}$$

is only a lower bound on the integral of the Fisher-Rao norm along the weight trajectory. We therefore make some additional assumptions in this section to draw out a crisp link between the Fisher-Rao *distance* and generalization gap along the trajectory.

Let $\ell(w; x, y) = -\log p_w(y|x)$ be the cross-entropy loss on sample (x, y). We assume that at each moment $\tau \in [0, 1]$, our model $p_{w(\tau)}(y|x)$ predicts on the interpolating distribution $p_\tau(y|x)$ well, that is

$$p_{w(\tau)}(y|x) \approx p_\tau(y|x)$$

for all input x; this is a reasonable assumption and corresponds to taking a large number of mini-batch updates in (12c). We approximate the FIM using the empirical FIM, i.e., we approximate the distribution $p_\tau(y|x)$ as a Dirac-delta distribution on the interpolated labels $y_\tau(x)$. Observe that

$$\begin{aligned}
\left\langle \dot{w(\tau)}, g(w(\tau))\dot{w(\tau)} \right\rangle &= \left\langle \dot{w(\tau)}, \mathop{\mathbb{E}}_{y|x\sim p_\tau} \partial_w \ell_{w(\tau)}(y|x) \partial_w \ell_{w(\tau)}(y|x)^\top \dot{w(\tau)} \right\rangle \\
&\approx \left\langle \dot{w(\tau)}, \partial_w \ell(w(\tau); x, y_\tau(x))\, \partial_w \ell(w(\tau); x, y_\tau(x))^\top \dot{w(\tau)} \right\rangle \\
&= \left| \frac{\ell(w(\tau+\mathrm{d}\tau); x, y_\tau(x)) - \ell(w(\tau); x, y_\tau(x))}{\mathrm{d}\tau} \right|^2 \\
&= \left| \frac{\Delta\ell(w(\tau))}{\mathrm{d}\tau} \right|^2,
\end{aligned} \tag{21}$$

where we use the shorthand

$$\Delta\ell(w(\tau)) := \ell(w(\tau+\mathrm{d}\tau); x, y_\tau(x)) - \ell(w(\tau); x, y_\tau(x)),$$

and plug (21) in the integration in (20)

$$\begin{aligned}
\int_0^1 \mathop{\mathbb{E}}_{x\sim p_\tau(x)} \left[\sqrt{2\mathrm{KL}\left(p_{w(\tau)}(\cdot|x),\ p_{w(\tau+\mathrm{d}\tau)}(\cdot|x)\right)}\right] \mathrm{d}\tau &= \int_0^1 \mathop{\mathbb{E}}_{x\sim p_\tau(x)} \sqrt{\left\langle \dot{w(\tau)}, g(w(\tau))\dot{w(\tau)} \right\rangle}\, \mathrm{d}\tau \\
&\approx \int_0^1 \mathop{\mathbb{E}}_{x\sim p_\tau(x)} \left[|\Delta\ell(w(\tau)| \right].
\end{aligned} \tag{22}$$

On the other hand, for moment τ let $\Omega_\tau \ni w(\tau)$ be a compact neighborhood of $w(\tau)$ in weights space, Rademacher complexity of the class of loss function is upper bounded as following

$$\begin{aligned}
\mathcal{R}_N(\Omega_\tau) &= \mathop{\mathbb{E}}_{\hat{p}\sim p_\tau^N} \mathop{\mathbb{E}}_{\sigma} \left[\sup_{w\in\Omega_\tau} \frac{1}{N} \sum_{i=1}^N \sigma^i \ell(w; x^i, y^i) \right] \\
&= \mathop{\mathbb{E}}_{\hat{p}\sim p_\tau^N} \mathop{\mathbb{E}}_{\sigma} \left[\sup_{w\in\Omega_\tau} \frac{1}{N} \sum_{i=1}^N \sigma^i \ell(w(\tau); x^i, y^i) + \sigma^i \left(\ell(w; x^i, y^i) - \ell(w(\tau); x^i, y^i) \right) \right] \\
&\leq \mathop{\mathbb{E}}_{\hat{p}\sim p_\tau^N} \mathop{\mathbb{E}}_{\sigma} \left[\frac{1}{N} \sum_{i=1}^N \sigma^i \ell(w(\tau); x^i, y^i) + \sup_{w\in\Omega_\tau} \frac{1}{N} \sum_{i=1}^N \left| \ell(w; x^i, y^i) - \ell(w(\tau); x^i, y^i) \right| \right], \\
&= 0 + \mathop{\mathbb{E}}_{\hat{p}\sim p_\tau^N} \left[\sup_{w\in\Omega_\tau} \frac{1}{N} \sum_{i=1}^N \left| \ell(w; x^i, y^i) - \ell(w(\tau); x^i, y^i) \right| \right] \\
&\longrightarrow \sup_{w\in\Omega_\tau} \mathop{\mathbb{E}}_{x\sim p_\tau} \left| \ell(w; x, y_\tau(x)) - \ell(w(\tau); x, y_\tau(x)) \right|
\end{aligned} \tag{23}$$

as N goes to infinity. The last step in (23) is followed by the compactness of Ω_τ and the Lipschitz continuity of the loss function. Let

$$\Omega_\tau := \{w| \mathop{\mathbb{E}}_{x\sim p_\tau} |\ell(w;x,y_\tau(x)) - \ell(w(\tau);x,y_\tau(x))| \le \mathop{\mathbb{E}}_{x\sim p_\tau} |\ell(w(\tau+\mathrm{d}\tau);x,y_\tau(x)) - \ell(w(\tau);x,y_\tau(x))|\}, \tag{24}$$

be the neighborhood of $w(\tau)$ within which the loss function changes less than $|\Delta\ell(w(\tau))|$. Compare this with (22), the Rademacher complexity of Ω_τ is exactly upper bounded by integration increments appearing in the expression for the Fisher-Rao distance. If we substitute $\|w(\tau)\|_{\text{FR}}$-ball in (15) with this modified Ω_τ, we have the following theorem.

Theorem 7. *Given a trajectory of the weights $\{w(\tau)\}_{\tau\in[0,1]}$ and a sequence $0 = \tau_0 \le \tau_1 < \tau_2 < ... < \tau_K \le 1$, for all $\epsilon > 2\sum_{k=1}^{K}(\tau_k - \tau_{k-1})\,\mathbb{E}_{x\sim p_\tau}|\Delta\ell(w(\tau_{k-1}))|$, the probability that*

$$\frac{1}{K}\sum_{k=1}^{K}\left(\mathop{\mathbb{E}}_{(x,y)\sim p_{\tau_k}}[\ell(\omega(\tau_k),x,y)] - \frac{1}{N}\sum_{(x,y)\sim\hat{p}_{\tau_k}}\ell(\omega(\tau_k),x,y)\right)$$

is greater than ϵ is upper bounded by

$$\exp\left\{-\frac{2K}{M^2}\left(\epsilon - 2\sum_{k=1}^{K}(\tau_k-\tau_{k-1})\mathop{\mathbb{E}}_{x\sim p_{\tau_k}}[|\Delta\ell(w(\tau_{k-1}))|]\right)\right\}. \tag{25}$$

Proof. The proof is same as in (15) except for substituting $\mathcal{R}_N(\|w(\tau_k)\|_{\text{FR}})$ with $\mathcal{R}_N(\Omega_{\tau_k})$ and using upper bounds (23), and

$$\begin{aligned}\Omega_{\tau_k} = \{w| &\mathop{\mathbb{E}}_{x\sim p_{\tau_k}}|\ell(w;x,y_{\tau_k}(x)) - \ell(w(\tau_k);x,y_{\tau_k}(x))| \\ &\le K(\tau_k-\tau_{k-1})\mathop{\mathbb{E}}_{x\sim p_{\tau_k}}|\ell(w(\tau_k);x,y_{\tau_k}(x)) - \ell(w(\tau_{k-1});x,y_{\tau_k}(x))|\}.\end{aligned} \tag{26}$$

□

We can now relate the Fisher-Rao distance (20) and the generalization bound in Thm. 7. For instance, if $\left|\frac{\mathrm{d}}{\mathrm{d}\tau}\ell(w(\tau);x,y_\tau(x))\right|$ is Riemann integrable over τ, then as K goes to infinity, there exists a sequence $0 = \tau_0 \le \tau_1 < \tau_2 < ... < \tau_K \le 1$ such that

$$\begin{aligned}\sum_{k=1}^{K}(\tau_k-\tau_{k-1})\mathop{\mathbb{E}}_{x\sim p_{\tau_k}}|\ell(w(\tau_k);x,y_{\tau_k}(x)) - \ell(w(\tau_{k-1});x,y_{\tau_k}(x))| &\longrightarrow \int_0^1 \mathop{\mathbb{E}}_{x\sim p_\tau(x)}|\ell(w(\tau+\mathrm{d}\tau);x,y_\tau(x)) - \ell(w(\tau);x,y_\tau(x))| \\ &\approx \int_0^1 \mathop{\mathbb{E}}_{x\sim p_\tau(x)}\left[\sqrt{2\mathrm{KL}\left(p_{w(\tau)}(\cdot|x),\ p_{w(\tau+\mathrm{d}\tau)}(\cdot|x)\right)}\right]\mathrm{d}\tau.\end{aligned} \tag{27}$$

This shows that computing the Fisher-Rao distance between two points on the statistical manifold results in a weight trajectory that minimizes the the generalization gap of weights trained on the interpolated distribution along the trajectory. In other words, one may either think of our coupled transfer process as computing the Fisher-Rao distance or as finding a weight trajectory that connects weights with a small generalization gap.

D. Frequently Asked Questions (FAQs)

1. **How is this distance better than methods such as Wasserstein distance, Maximum Mean Discrepancy (MMD), Hellinger distance or other f-divergences to measure distances between probability distributions?**

   Measuring distance between learning tasks is different than measuring distances between the respective data distributions. The above concepts can only measure distances between data distributions, they do not consider the hypothesis class used to transfer across the two distributions and therefore do not reflect the true difficulty of transfer. The experiment in Fig. 7 demonstrates this. This point in fact is the central motivation of our paper. Also see the discussion of related work in Sec. 6.

2. **Why do your distances range from small to large values?**

We discuss this in Rem. 4. The scale of distances can be quite different for different hypothesis spaces but this is not a problem if they can be compared across architectures for the same task pair. Since the coupled transfer distance measures the length of the trajectory on the statistical manifold which is invariant to the specific parameterization of the model, the numerical value of the distance has a sound grounding in theory and not on some arbitrary scale. Further, just like the cosine distance scales with the inner product and can be normalized using the ℓ_2 norm of the respective vectors, we envision that our distance can be normalized using the coupled transfer distance to some "canonical" task (say, actual vs. fake source/target images) to get a better dynamic range. We are currently studying which tasks are good canonical tasks for this purpose.

3. **The coupled transfer distance trains the model multiple times between source and target tasks to estimate the distance. How is this useful in practice to select, say, a good source dataset to pre-train from? Interesting formulation, but too complex to use in practice.**

   We think of our work as a first step towards the challenging problem of understanding distances between learning tasks. Our final goal is indeed to use the tools developed here for practical applications, e.g., to design methods that can select the best source task to transfer from while fitting a given task or the best architecture to transfer between a given set of tasks, but we are not there yet. The practical utility of this work is to identify that typical methods in the literature for measuring task distances (see related work discussed in Sec. 6) leave a lot on the table. Theoretically they do not explicitly characterize the hypothesis class being transferred. Empirically, distances estimated by typical methods do not correlate strongly with the difficulty of fine-tuning (see Figures 2 and 3). Our development provides concrete theoretical tools to understand other task distances that correlate well with the coupled transfer distance, and thereby the difficulty of fine-tuning.

   For the same reason, we do not think the technical complexity of formalizing and computing the coupled transfer distance should take anything away from its intellectual metric. Our goal is to develop theoretical tools to understand when transfer between tasks is easy and when it is not, it is not to develop a good fine-tuning algorithm.

4. **Does coupled transfer obtain better generalization error on the target task than standard fine-tuning?**

   Coupled transfer explicitly modifies the task while standard fine-tuning does not, so this is a natural question. We have explored it in Fig. 6. Our experiment shows that, broadly, the coupled transfer improves generalization. This is consistent with existing literature, e.g., Gao & Chaudhari (2020), which employs task interpolation for better transfer learning. We however note that improving fine-tuning is not our goal in this paper; in fact, we want our task distance to correlate with the difficulty of fine-tuning.

5. **Feature extractor φ for initializing Γ^0 is trained on a generic task, how is this task related to source/target?**

   We discuss this on Lines 198–215 (right column) in the main paper. The feature extractor is only used to initialize the coupling Γ^0, couplings in successive iterations Γ^k are computed using the ground metric in (12b) and do not use the feature extractor. Using a feature extractor to compute OT distances is quite common in the literature, e.g., (Cui et al., 2018). We use a ResNet-50 pre-trained on ImageNet as the feature generator to compute the initialization Γ^0 for all experiments in this paper.

   We also performed some experiments where Γ^0 was initialized using the quadratic ground metric $C_{ij} = \|x_s^i - x_t^j\|_2^2$ without using the feature generator. The coupled task distance converged to 0.18 for MNIST-CIFAR-10 and 0.062 in the other direction after $k = 5$ iterations; this is the same as the case when the coupling is initialized using the feature generator as mentioned on Lines 324–327 (right column) in the main paper.

   ImageNet is a different task than the ones considered in this paper (subsets of MNIST, CIFAR-10, CIFAR-100 and Deep Fashion). If the feature generator's task is closely related to only one of the source/target tasks but not the other, the task distance will require more iterations to converge. For our experimental setup, ImageNet is, roughly speaking, a superset of the tasks we analyze, this enables the coupled transfer distance in our experiments to converge within 4–5 iterations. Note that each iteration of (12) is quite non-trivial and takes a few GPU-hours; it performs multiple epochs of weight updates and estimates C_{ij} along the trajectory to update all the blocks of the coupling matrix Γ^k.

6. **The expression for the interpolated distribution in ?? is for the quadratic ground metric $C_{ij} = \|x_s^i - x_t^j\|_2^2$ but the ground metric in (12b) is different.**

   The interpolation in **??** McCann's displacement convexity (McCann, 1997) for the space of probability measures under the Wasserstein metric. This result identifies when functionals on the space of probability measures are convex along geodesics. More formally, if $F : \mathcal{P}(\Omega) \to \mathbb{R}$ is λ-geodesically-convex functional, then

$$(1-\tau)F(\rho_0) + tF(\rho_1) \geq F(\rho_\tau) + \frac{\lambda}{2}\tau(1-\tau)W_2(\rho_0, \rho_1)^2;$$

here $\rho_0, \rho_1 \in \mathcal{P}(\Omega)$ are two probability measures supported on the set Ω and ρ_τ is the interpolant at time τ along the geodesic in W_2 metric joining them. Computing displacement interpolation for general ground metrics, even analytically, is difficult; see Villani (2008, Chapters 16–17). It is therefore very popular in the optimal transport literature to study interpolation under the quadratic ground metric. In order to keep the implementation simple and focus on the main idea of coupled transfer, we use the expression for displacement interpolation p_τ in **??** for the quadratic ground metric $C_{ij} = \|x_s^i - x_t^j\|_2^2$ but compute the optimal coupling Γ using the Fisher-Rao distance $C_{ij} = d_{\text{FR}}(p_{w(0)}(\cdot \mid x_s^i), p_{w(1)}(\cdot \mid x_t^j))$ as the tasks are interpolated using the coupling of the previous iteration Γ^{k-1}; see (12b). Note that this does not change the fact that p_τ is *an interpolation*, it is however not a displacement interpolation anymore for our particular chosen ground metric C_{ij}. This is a pragmatic choice which keeps our theoretical development tractable.

7. **Why use Beta$(\tau, 1-\tau)$ to interpolate?**

   We discuss this on Lines 217–228 in the main paper. Mathematically, employing this technique really means that we use some other ground metric than the quadratic cost in the OT problem; this is a minor modification with a big benefit of keeping the interpolated task within the manifold of natural images.

8. **How do you compute the integral in (12b)?**

   Integral on τ in (12b) is computed using its Riemann approximation along the weight trajectory $\{w(\tau) : t \in [0,1]\}$ given by (12c).

9. **Why do Thm. 5 and Thm. 6 do not use the standard PAC-learning analysis?**

   PAC analysis without ground-truth labels for the data from the interpolated distribution is difficult. We therefore bound the generalization gap in terms of the loss $\ell(w, x, y)$ where the label generating mechanism is a simple linear interpolation between one-hot labels of the source and target tasks. Let us note that a PAC-Bayes bound between the source and target posterior weight distributions is given in Achille et al., 2019c.

10. **Why should a larger model have a smaller coupled transfer distance in Fig. 5 compared to Fig. 3?**

    We discuss this on Lines 374–383 in the main paper.