A ADDITIONAL TRANSPORTATION MAPPINGS DETAILS AND PROOFS

A.1 FINDING K-PAIRED CLUSTERS

Algorithm 1 Finding k Paired Clusters

```
**Input:** $X, Y, k$
$d \leftarrow X.ndim$
$T_{OT} \leftarrow$ OptimalTransportAlg$(X, Y)$ //e.g., Sinkhorn
$Z \leftarrow [X, T_{OT}(X)]$
$Z_{cluster-centroids} \leftarrow$ ClusteringAlg$(Z, k)$ //e.g., k-means
$M_{src} \leftarrow [Z_{cluster-centroids}]_{1:d}$ //slicing column-wise
$M_{tgt} \leftarrow [Z_{cluster-centroids}]_{d:2d}$
**Output:** $M_{src}$, $M_{tgt}$
```

A.2 INFINITE NUMBER OF POSSIBLE MAPPINGS BETWEEN DISTRIBUTIONS

As stated in the introduction, given two distributions, there exist many possible mappings such that $T_\sharp P_{src} = P_{tgt}$ (it should be noted that here we are speaking of the general mapping problem, not the *optimal* transport problem which can be shown via Brenier's theorem Peyré & Cuturi (2019) to have a unique matching for some cases). For instance, given two isometric Gaussian distributions $\boldsymbol{x} \sim \mathcal{N}_1(\mu_1, I)$, $\boldsymbol{y} \sim \mathcal{N}_2(\mu_2, I)$, where I is the Identity matrix, there exist an infinite number of T's such that $T(\boldsymbol{x}) \sim \mathcal{N}_2$. Specifically, any T of the form: $T(\boldsymbol{x}) = \mu_2 + R(\boldsymbol{x} - \mu_1)$, where is R is an arbitrary rotation matrix, will shift $T_\sharp \mathcal{N}_1$ to have a mean of μ_2 and perfectly align the two distributions (since any rotation of an isometric Gaussian will still be an isometric Gaussian).

A.3 DISTANCE IN EMPIRICAL INTERPRETABLE TRANSPORT UPPER-BOUNDS THE WASSERSTEIN DISTANCE

First, let's remember our empirical method for finding T:

$$\arg\min_{T \in \Omega} \frac{1}{N} \sum_{i}^{N} c(\boldsymbol{x}^{(i)}, T(\boldsymbol{x}^{(i)})) + \lambda d(T(\boldsymbol{x}^{(i)}), T_{OT}(\boldsymbol{x}^{(i)})) \tag{5}$$

where T_{OT} is the optimal transport solution between our source and target domains with the given c cost function. The distance term d on the right-hand side of this equation is assumed to be the ℓ_2 cost or squared euclidean cost, and is an empirical approximation of the divergence term $\phi(P_{T(\boldsymbol{x})}, P_Y)$ in Eqn. 1, where ϕ is assumed to be the Wasserstein distance, W. We claim this is a reasonable approximation since as N approaches the size of the dataset (or for densities, $\lim_{N\to\infty}$), the distance term becomes the expectation: $\mathbb{E}_{x\sim P_{src}} d(T(\boldsymbol{x}^{(i)}), T_{OT}(\boldsymbol{x}^{(i)}))$ which is an upper-bound on the $W(P_{T(\boldsymbol{x})}, P_Y)$ distance. To show this, we start with the expanded W distance:

$$\begin{aligned}
W(P_{T(\boldsymbol{x})}, P_Y) &= \min_{R\in\Psi} \mathbb{E}_{\boldsymbol{x}\sim P_{src}} d\left(T(\boldsymbol{x}), R(T(\boldsymbol{x}))\right), \quad && \Psi := \{R : R_\sharp T(\boldsymbol{x}) = P_Y\} \\
&\leq \mathbb{E}_{\boldsymbol{x}\sim P_{src}} d\left(T(\boldsymbol{x}), R(T(\boldsymbol{x}))\right), && \forall R \in \Psi \\
&\text{If we let } Q = T_{OT} \cdot T^{-1}\text{, and since } Q \in \Psi \text{ we can say} \\
&\leq \mathbb{E}_{\boldsymbol{x}\sim P_{src}} d\left(T(\boldsymbol{x}), Q(T(\boldsymbol{x}))\right) = \mathbb{E}_{\boldsymbol{x}\sim P_{src}} d\left(T(\boldsymbol{x}), T_{OT}(\boldsymbol{x})\right) \\
\implies W(P_{T(\boldsymbol{x})}, P_Y) &\leq \mathbb{E}_{\boldsymbol{x}\sim P_{src}} d\left(T(\boldsymbol{x}), T_{OT}(\boldsymbol{x})\right)
\end{aligned}$$

A.4 PROVING THE K-SPARSE OPTIMAL TRANSPORT IS THE K-SPARSE TRANSPORT THAT MINIMIZES OUR DISTANCE FROM OT LOSS

When performing unrestricted k-sparse transport, i.e. where $\Omega^{(k)}_{sparse}$ is any transport which only moves points along k dimensions, the k-sparse optimal transport solution is the exact mapping that minimizes the distance function in the right-hand side of Eqn. 5 if d is the ℓ_2 distance or squared Euclidean distance. As a reminder, k-sparse optimal transport is: $[T(\boldsymbol{x})]_j = [T_{OT}(\boldsymbol{x})]_j$ if $j \in \mathcal{A}$, else $[\boldsymbol{x}]_j$, where $\mathcal{A}$ is the active set of k dimensions which our k-sparse transport T can move points. Let $\bar{\mathcal{A}}$ be $\mathcal{A}$'s compliment (i.e. the dimensions which are unchanged under T). Let $\boldsymbol{z} = T(\boldsymbol{x})$, $\boldsymbol{z}_{OT} = T_{OT}(\boldsymbol{x})$, and $\boldsymbol{x} \in \mathbb{R}^{n \times d}$. If d is the squared Euclidean distance:

$$\begin{aligned}
d(\boldsymbol{z}, \boldsymbol{z}_{OT}) &= \sum_{j \in [d]} \sum_{i \in [n]} \left(\boldsymbol{z}_{i,j} - \boldsymbol{z}_{OT_{i,j}}\right)^2 \\
&= \sum_{j \in \mathcal{A}} \sum_{i \in [n]} \left(\boldsymbol{z}_{i,j} - \boldsymbol{z}_{OT_{i,j}}\right)^2 + \underbrace{\sum_{j \in \bar{\mathcal{A}}} \sum_{i \in [n]} \left(\boldsymbol{x}_{i,j} - \boldsymbol{z}_{OT_{i,j}}\right)^2}_{=\alpha \text{ , since constant w.r.t T}} \\
&= \sum_{j \in \mathcal{A}} \sum_{i \in [n]} \left(\boldsymbol{z}_{i,j} - \boldsymbol{z}_{OT_{i,j}}\right)^2 + \alpha \\
&\text{now if T is the truncated optimal transport solution, } [\boldsymbol{z}]_j = [\boldsymbol{z}_{OT}]_j \quad \forall j \in \mathcal{A} \\
&= 0 + \alpha
\end{aligned}$$

Since α is the minimum of $d(\boldsymbol{z} - \boldsymbol{z}_{OT})$ for a given $\mathcal{A}$, the truncated optimal transport problem minimizes the $d(T(\boldsymbol{x}^{(i)}), T_{OT}(\boldsymbol{x}^{(i)}))$ distance. This can easily be extended to show that the optimal active set for this case is the one that minimizes α, thus the active set should be the k dimensions which have the largest squared difference between $\boldsymbol{x}$ and $\boldsymbol{z}_{OT}$.

A.5 PROOF THAT K-MEAN SHIFT IS THE K-VECTOR SHIFT THAT GIVES US THE BEST ALIGNMENT

When performing k-sparse vector transport, i.e. where $\Omega^{(k)}_{vector} = \{T : T(\boldsymbol{x}) = \boldsymbol{x} + \tilde{\delta}\}$ where $\tilde{\delta} = [\delta]_j$ if $j \in \mathcal{A}$ else $[\delta]_j = 0$ and $\delta \in \mathbb{R}^d$, $|\mathcal{A}| \leq k$, the k-sparse mean shift solution is the exact mapping that minimizes the distance function in the right-hand side of Eqn. 5 when d is the ℓ_2 distance.

$$\begin{aligned}
d(\boldsymbol{z}, \boldsymbol{z}_{OT}) &= \sum_{j \in [d]} \sum_{i \in [n]} \left(\boldsymbol{z}_{i,j} - \boldsymbol{z}_{OT_{i,j}}\right)^2 \\
&= \sum_{j \in \mathcal{A}} \sum_{i \in [n]} \left(\boldsymbol{z}_{i,j} - \boldsymbol{z}_{OT_{i,j}}\right)^2 + \underbrace{\sum_{j \in \bar{\mathcal{A}}} \sum_{i \in [n]} \left(\boldsymbol{x}_{i,j} - \boldsymbol{z}_{OT_{i,j}}\right)^2}_{=\alpha \text{ , since constant w.r.t T}} \\
&= \sum_{j \in \mathcal{A}} \sum_{i \in [n]} \left(\boldsymbol{z}_{i,j} - \boldsymbol{z}_{OT_{i,j}}\right)^2 + \alpha \\
&= \sum_{j \in \mathcal{A}} \sum_{i \in [n]} \left(\boldsymbol{x}_{i,j} + \delta_j - \boldsymbol{z}_{OT_{i,j}}\right)^2 + \alpha \\
&= \sum_{j \in \mathcal{A}} \sum_{i \in [n]} \Big(\boldsymbol{x}^2_{i,j} + \delta_j^2 + \boldsymbol{z}^2_{OT_{i,j}} + 2\delta_j(\boldsymbol{x}_{i,j} - \boldsymbol{z}_{OT_{i,j}}) - 2\boldsymbol{z}_{OT_{i,j}}\delta_j - 2\boldsymbol{x}_{i,j}\boldsymbol{z}_{OT_{i,j}}\Big) + \alpha
\end{aligned}$$

Similar to the k-sparse optimal transport solution, we can see that $\mathcal{A}$ should be selected as the k dimensions which have the largest shift, thus minimizing α. The coordinate-wise gradient of the above equation is:

$$\nabla_{\delta_j} d(\boldsymbol{z}, \boldsymbol{z}_{OT}) = \begin{cases} \sum_{i \in [n]} \left(2\delta_j + 2\boldsymbol{x}_{i,j} - 2\boldsymbol{z}_{OT_{i,j}}\right) & j \in \mathcal{A} \\ 0 & j \in \bar{\mathcal{A}} \end{cases}$$

Now with this we can say:

$$\begin{aligned}
\nabla_{\delta_{j\in\mathcal{A}}} d(\boldsymbol{z}, \boldsymbol{z}_{OT}) &= \sum_{i\in[n]} \big(2\delta_j + 2\boldsymbol{x}_{i,j} - 2\boldsymbol{z}_{OT_{i,j}}\big) \\
&= 2n\delta_j + \sum_{i\in[n]} \big(2\boldsymbol{x}_{i,j} - 2\boldsymbol{z}_{OT_{i,j}}\big) \\
&\text{now let } \delta_j = \delta_j^* \\
0 &= 2n\delta_j^* + \sum_{i\in[n]} \big(2\boldsymbol{x}_{i,j} - 2\boldsymbol{z}_{OT_{i,j}}\big) \\
n\delta_j^* &= \sum_{i\in[n]} \big(\boldsymbol{z}_{OT_{i,j}} - \boldsymbol{x}_{i,j}\big) \\
\delta_j^* &= \frac{1}{n}\sum_{i\in[n]} \big(\boldsymbol{z}_{OT_{i,j}} - \boldsymbol{x}_{i,j}\big) \\
\delta_j^* &= \mu_{\boldsymbol{z}_{OT_j}} - \mu_{\boldsymbol{x}_j}
\end{aligned}$$

Thus showing the optimal delta vector to minimize k-vector transport is exactly the k-sparse mean shift solution.

B CHALLENGES OF EXPLAINING DISTRIBUTION SHIFTS AND LIMITATIONS OF OUR METHOD

Distribution shift is a ubiquitous, and quite challenging problem. Thus, we believe discussing the challenges of this problem and the limitations of our solution should aid in advancements in this area of explaining distribution shifts.

As mentioned in the main body, as distribution shifts can take many forms, trying to explain a distribution shift is a highly context-dependent problem (i.e., dependent on the data setting, task setting, and operator knowledge). Thus, a primary challenge in developing distribution shift explanations is determining how to evaluate whether a given explanation is valid for a given context. In this work, we hope to introduce the problem of explaining distribution shifts *in general* (i.e. not with a specific task nor setting in mind), therefore we do not have an automated way of measuring whether a given explanation is indeed interpretable. Evaluating explanations is an active area of research Robnik-vSikonja & Bohanec (2018); Molnar (2020); Doshi-Velez & Kim (2017) with commonalities such as an explanation should be contrastive, succinct, should highlight abnormalities, and should have high fidelity. Instead, we introduce a proxy method which supplies the operator with the Percent-Explained and the adjustable k-level of sparse/cluster mappings but leaves the task of validating the explanation up to the operator. We believe developing new shift explanation maps and criteria for a specific applications (e.g., explaining the results of experiments ran with different initial conditions) is a rich area for future work.

Explaining distribution shifts becomes more difficult when the original data is not interpretable. This typically can take two forms: 1) the raw data *features* are uninterpretable but the samples are interpretable (e.g., a sample from the CelebA dataset Liu et al. (2015) is interpretable but the pixel-level features are not) or 2) when both the raw data features and samples are uninterpretable (e.g., raw experimental outputs from material science simulations). In the first case, one can use the set of counterfactual pairs method outlined in subsection 6.1 (see Fig. 8 for examples with CelebA), however, as mentioned in the main paper, this is less sample efficient than an interpretable transport map. For the second case, if the original features are not interpretable, one must first find an interpretable latent feature space – which is a challenging problem by itself. As seen in Fig. 10, it is possible to solve for a semantic latent space and solve interpretable transport maps within the latent space, in this case, the latent space of a VAE model. However, if the meaningful latent features are not extracted, then any transport map within this latent space will be meaningless. In the case of Fig. 10, the 3-cluster explanation is likely only interpretable because we know the ground truth and thus know what to look for. As such, this is still an open problem and one we hope future work can improve on.

Additionally, while the PercentExplained metric shows the fidelity of an explanation (i.e. how aligned $T_\sharp(P_{src})$ and P_{tgt} are), we do not have a method of knowing specifically *what* is missing from the explanation. This missing part of the explanation can be considered a "known unknown". For example, if a given T has a PercentExplained of 85%, we know how much is missing, but we do not know what information is contained in the missing 15%. Similarly, when trying to explain an image-based distribution shift with large differences in content (e.g., a dataset with blonde humans and a dataset with bald humans), leveraging existing style transfer architectures (where one wishes to only change the style of an image while retaining as much of the original content as possible) to generate distributional counterfactuals can lead to misleading explanations. This is because explaining image-based distribution shifts might require large changes in content (such as removing head hair from an image), which most style-transfer models are biased against doing. As an example, Fig. 8 shows an experiment that translates between five CelebA domains (blond hair, brown hair, wearing hat, bangs, bald). It can be seen that the StarGAN model can successfully translate between stylistic differences such as "blond hair" $\rightarrow$ "brown hair" but is unable to make significant content changes such as "bangs" $\rightarrow$ "bald".

The above issues are mainly problems that affect distribution shift explanations in general, but below are issues specific to our shift explanation method (or any method which similarly uses empirical OT). Since we rely on the empirical OT solution for the sparse and cluster transport (and the percent explained metric), the weaknesses of empirical OT are also inherited. For example, empirical OT, even using the Sinkhorn algorithm with entropic regularization, scales at least quadratically in the number of samples Cuturi (2013). Thus, this is only practical for thousands of samples. Furthermore, empirical OT is known to poorly approximate the true population-level OT in high dimensions although entropic regularization can reduce this problem Genevay et al. (2019). Finally, empirical OT does not provide maps for new test points. Some of these problems could be alleviated by using recent Wasserstein-2 approximations to optimal maps via gradients of input-convex neural networks based on the dual form of Wasserstein-2 distance Korotin et al. (2019); Makkuva et al. (2020). Additionally, when using k-cluster maps, the clusters are not guaranteed to be significant (i.e. it might be indiscernible what makes this cluster different than another cluster), and thus if using k-cluster maps on datasets which do not have natural significant clusters (e.g., P_{src} $\sim$Uniform$(0, 1)$, P_{tgt} $\sim$Uniform$(1, 2)$) an operator might waste time looking for significance where there is none. While this cannot be avoided in general, using a clustering method which is either specifically designed for finding interpretable clusters Fraiman et al. (2013); Bertsimas et al. (2021) or one which directly optimizes the objective in interpretable transport equation Eqn. 1 might lead to easier to explanations which are easier to interpret or validate.

C EXPERIMENTS ON KNOWN SHIFTS

Here we present additional result on simulated experiments as well as an experiment on UCI "Breast Cancer Wisconsin (Original)" dataset Mangasarian & Wolberg (1990). Our goal is to illuminate when to use the different sets of interpretable transport, and how the explanations can be interpreted, where in this case, a ground truth explanation is known. [1]

C.1 SIMULATED EXPERIMENTS

In this section we study three toy shift problems: a mean shift between two, otherwise identical, Gaussian distributions, a Gaussian mixture model where each mixture component has a different mean shift, and a flipped and shifted half moon dataset, as seen in figures (a), (d), and (g) in Fig. 4.

The first case is of a mean shift between two, otherwise identical, Gaussian distributions can be easily explained using k-sparse mean shift (as well as vanilla mean shift). We first calculate the OT mapping T_{OT} between the two Gaussian distributions, which has a closed form solution of $T_{OT}(\boldsymbol{x}) = \mu_{tgt} + A(\boldsymbol{x} - \mu_{src})$, where A is a matrix that can be seen as a conversion between the source and target covariance matrices, and because the covariance matrices are identical, A is the identity.

---

[1] Code to recreate all experiments seen here and in the main body of the paper will be released upon acceptance.

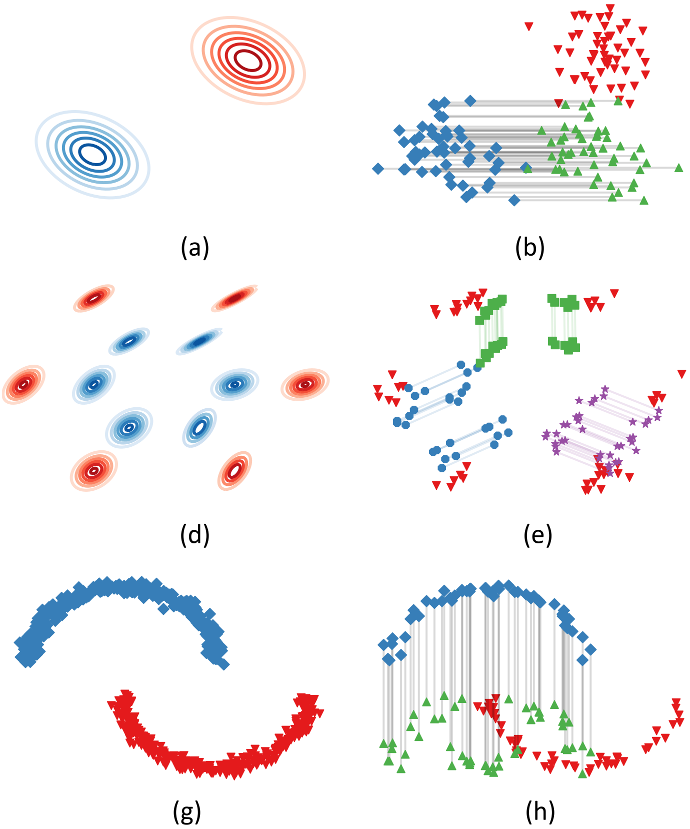


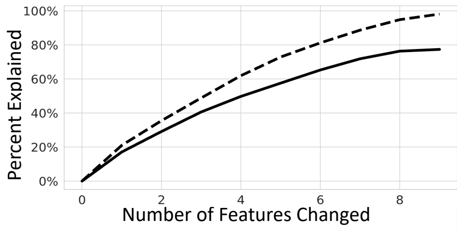
Figure 4: Three toy dataset shift examples showing the advantages of the different shift explanation methods, where a mean shift between Gaussians (top row) can be easily explained using k-sparse vector shifts, a varying mean shift across mixture components of a Gaussian mixture model (middle row) is best explained using k-sparse transport maps, while a complex shift (bottom row) requires a complex feature-wise mapping, such as k-sparse optimal transport, which maximally aligns the distributions, at the expense of interpretability. Each example shows three levels of decreasing interpretability, where the leftmost column shows the original shift (which has maximal interpretability since $k = 0$) from source (blue diamonds) to target (red down arrows), and the rightmost column shows an shift with near perfect fidelity.

The second toy example of distribution shift is a shifted Gaussian mixture model which represents a case where groups within a distribution shift in different ways. An example of this type of shift could be explaining the change in immune response data across patients given different forms of treatment for a disease. Looking at (d) in Fig. 4, it is clear that sparse feature transport will not easily explain this shift. Instead, we turn to cluster-based explanations, where we first find k paired clusters and attempt to show how these shift from P_{src} to P_{tgt}. Following the mean-shift transport of paired clusters approach outlined in subsection 4.4, the $k = 3$ case as seen in the Appendix shows that three clusters can sufficiently approximate the shift by averaging the shift between similar groups. If a more faithful explanation is required, (f) of Fig. 4 shows that increasing k to 6 clusters can recover the full shift, i.e. PercentExplained=100, at the expense of being less interpretable (which is especially true in a real-world case where the number of dimensions might be large).

The half moon example, figure (g) in Fig. 4, shows a case where a complex feature-wise dependency change has occurred. This example is likely best explained via feature-wise movement, so will use k-sparse transport. If we follow the approach in subsection 4.3 with our interpretable set as the $\Omega^{(k)}$ and let $k = 1$, we get a mapping which is interpretable, but has poor alignment (see Figure (h) in Fig. 4). For this example, we can possibly reject this explanation due to a poor PercentExplained. With the understanding that this shift is not explainable via just one feature, we can instead use a

$k = 2$-sparse OT solution. The $k = 2$ case can be seen in (i) of Fig. 4 which shows that this approach aligns the distributions perfectly, at the expense of interpretability.

C.2 EXPLAINING SHIFT IN WISCONSIN BREAST CANCER DATASET



2 Features: "Bare Nuclei and Cell Size were shifted by 6.28 and 5.27"

5 Features: "Bare Nuclei, Cell Size, Cell Shape, Normal Nucleoli, and Marginal Adhesion were shifted by 6.28, 5.27, 5.15, 4.6, 4.24"

All Features: "Bare Nuclei, Cell Size, Cell Shape, Normal Nucleoli, Marginal Adhesion, Clump Thickness, Bland Chromatin, Single Epithelial Cell Size, and Mitoses were shifted by 6.28, 5.27, 5.15, 4.6, 4.24, 4.22, 3.89, 3.22, 1.54

Figure 5: A comparison of the performance of k-sparse mean shift explanations (solid line) and k-sparse optimal transport explanations (dashed line) when explaining the shift from the benign tumor samples to malignant tumor samples for the UCI Wisconsin Breast Cancer dataset. On the right are example explanations a human operator would see as they change the level of interpretability during k-sparse mean shift explanations (where "All Features" is the baseline full mean shift explanation).

This tabular dataset consists of tumor samples collected by Mangasarian & Wolberg (1990) where each sample is described using nine features which are normalized to integers from $[0, 10]$. We split the dataset along the class dimension and set P_{src} to be the 443 benign tumors and P_{tgt} as the 239 malignant samples. To explain the shift, we used two forms of k-sparse transport, the first being k-sparse mean transport and the second being k-sparse optimal transport. The left of Fig. 5 shows that the k-sparse mean shift explanation is sufficient for capturing the 50% of the shift between P_{src} and P_{tgt} using only four features, and nearly 80% of the shift with all 9 features. However, if an analyst requires a more faithful mapping, they can use the k-sparse OT explanation which can recover the full shift, at the expense of the interpretability. The right of Fig. 5 shows example explanations which an analyst can use along with their context-specific expertise for determining the main differences between the different tumors they are studying.

C.3 COUNTERFACTUAL EXAMPLE EXPERIMENT TO EXPLAIN A MULTI-MNIST SHIFT

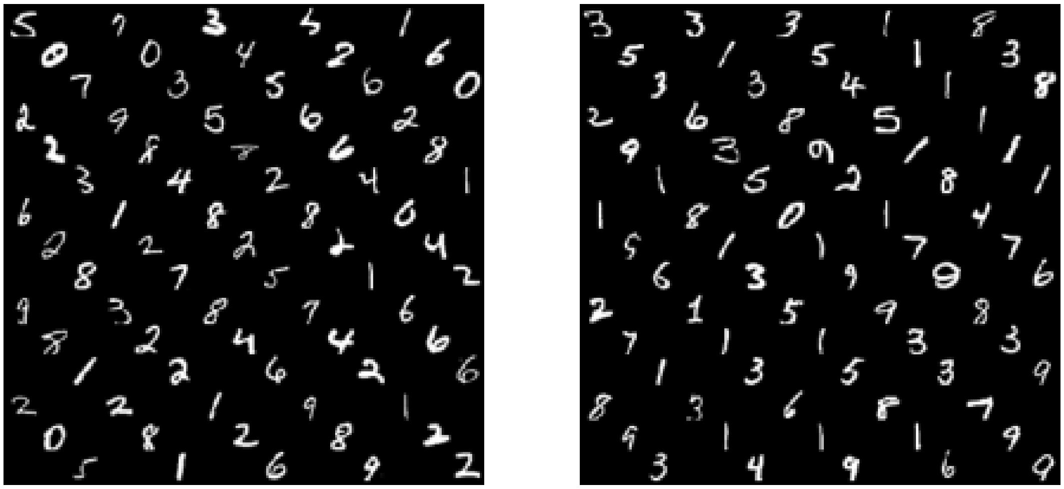


Figure 6: A grid of 25 raw samples from each domain (left is P_{src} and right is P_{tgt}). Even for the relatively simple shift seen in the Shifted Multi-MNIST dataset, it may be hard to tell what is different between the two distributions by just looking at samples (without knowing the oracle shift).

As mentioned in subsection 6.1, image-based shifts can be explained by supplying an operator with a set of distributional counterfactual images with the notion that the operator would resolve which semantic features are distribution-specific. Here we provide a toy experiment (as opposed to the real world experiment seen in subsection 6.1) to illustrate the power of distributional counterfactual examples. To do this, we apply the distributional counterfactual example approach to a Multi-MNIST dataset where each sample consists of a row of three randomly selected MNIST digits Deng (2012) and is split such that P_{src} consists of all samples where the middle digit is even and zero and P_{tgt} is all samples where the middle digit is odd, as seen in Fig. 6.

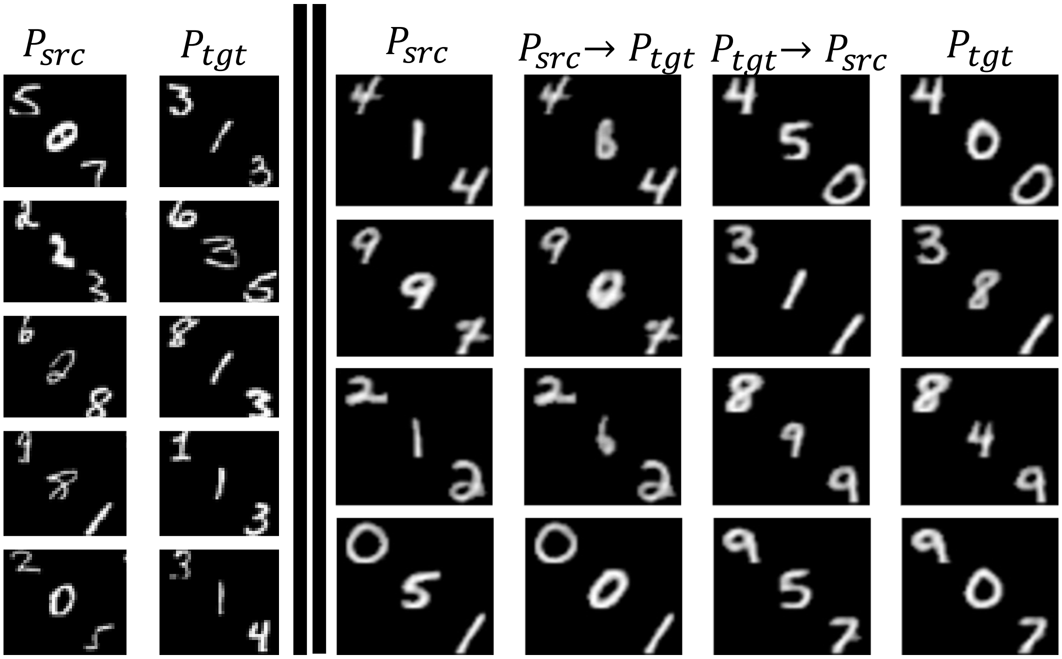


Figure 7: A comparison of the baseline grid of unpaired source and target samples (left) and counterfactual pairs (right) which shows how counterfactual examples can highlight the difference between the two distributions. For each image, the top left digit represents the class label, the middle digit represents the distribution label (where P_{src} only contains even digits and zero and P_{tgt} has odd digits), and the bottom right digit is noise information and is randomly chosen. The second, third columns show the counterfactuals from $P_{src} \rightarrow P_{tgt}$ and $P_{tgt} \rightarrow P_{src}$, respectively. Hence we can see under the push forward of each image the "evenness" of the domain digit changes while the class and noise digits remain unchanged.

Algorithm 2 Generating distributional counterfactuals using DIVA

```
Input: $\boldsymbol{x}_1 \sim D_1$, $\boldsymbol{x}_2 \sim D_2$, model
$z_{y_1}, z_{d_1}, z_{residual_1} \leftarrow$ model.encode($\boldsymbol{x}_1$)
$z_{y_2}, z_{d_2}, z_{residual_2} \leftarrow$ model.encode($\boldsymbol{x}_2$)
$\hat{\boldsymbol{x}}_{1\rightarrow 2} \leftarrow$ model.decode($z_{y_1}, z_{d_2}, z_{residual_1}$)
$\hat{\boldsymbol{x}}_{2\rightarrow 1} \leftarrow$ model.decode($z_{y_2}, z_{d_1}, z_{residual_2}$)
Output: $\hat{\boldsymbol{x}}_{1\rightarrow 2}$, $\hat{\boldsymbol{x}}_{2\rightarrow 1}$
```

To generate the counterfactual examples, we use a Domain Invariant Variational Autoencoder (DIVA) Ilse et al. (2020), which is designed to have three independent latent spaces: one for class information, one for domain-specific information (or in this case, distribution-specific information), and one for any residual information. We trained DIVA on the Shifted Multi-MNIST dataset for 600 epochs with a KL-β value of 10 and latent dimension of 64 for each of the three sub-spaces. Then, for each image counterfactual, we sampled one image from the source and one image from the target and encoded each image into three latent vectors: z_y, z_d, and $z_{residual}$. The latent encoding z_d was then "swapped" between the two encoded images, and the resulting latent vector set was decoded to produce the counterfactual for each image. This process is detailed in Algorithm 2 below. The resulting counterfactuals can be seen in Fig. 7 where the middle digit maps from the source (i.e., odd digits) to the target (i.e., even digits) and vice versa while keeping the other content unchanged (i.e., the top and bottom digits).

C.4 USING STARGAN TO EXPLAIN DISTRIBUTION SHIFTS IN CELEBA

Here we apply the distributional counterfactual approach seen in subsection 6.1 to the CelebA dataset Liu et al. (2015), which contains over 200K images of celebrities, each with 40 attribute annotations. We split the original dataset into 5 related sets, P_1="blonde hair", P_2="brunette hair", P_3="wearing hat", P_4="bangs", P_5="bald". These five sets were chosen as they are related concepts (all related to hair) yet mostly visually distinct. Although there are images with overlapping attributes, such as a blonde/brunette person with bangs, these are rare and naturally occurring, thus they were not excluded.

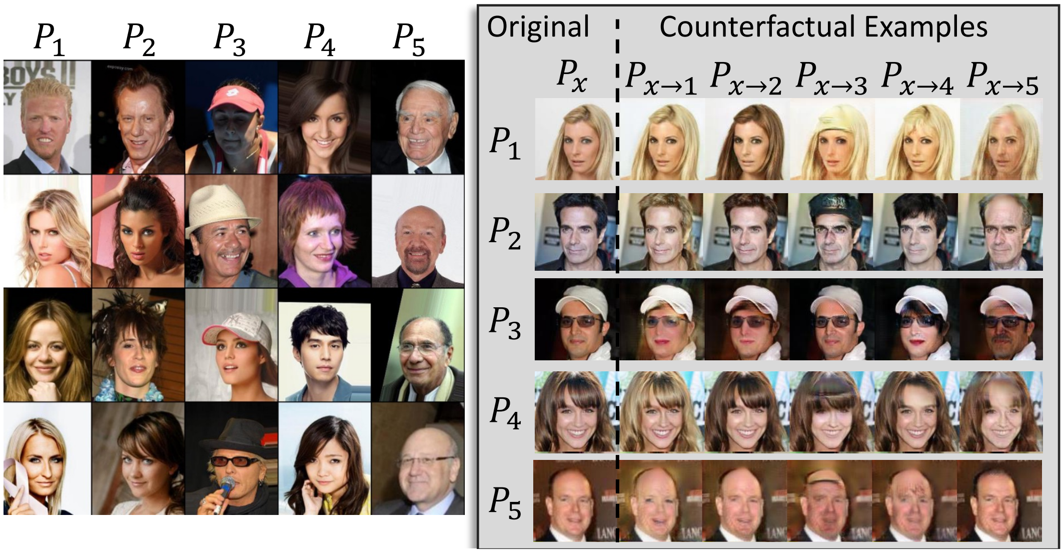


Figure 8: StarGAN is able to adequately translate between distributions with similar content but different style (e.g., $P_1 \to P_2$), however, when transporting between distributions with different content (e.g., "no hat" $\to P_3$) the I2I model is unable to properly capture the shift. This is likely due to the model being biased to only change the *style* of the image, while maintaining as much *content* as possible. The figure breakdown is similar to Fig. 3 with the baseline method of unpaired samples on the left and paired counterfactual images on the right, where here P_1="blonde hair", P_2="brunette hair", P_3="wearing hat", P_4="bangs", P_5="bald".

We trained a StarGAN model Choi et al. (2018) to generate distributional counterfactuals following the same approach seen in subsection 6.1. The result of this process can be seen in Fig. 8, where we can see the model successfully translating "stylistic" parts of the image such as hair color. However, the model is unable to translate between distributions with larger differences in "content" such as removing hair when translating to "bald". This highlights a difference between I2I tasks such as style transfer (where one wishes to bias a model to only change the style of an image while retaining as much of the original content as possible) the mappings required for explaining image-based distribution shifts, which might require large changes in content (such as adding a hat to an image).

D EXPLAINING SHIFTS IN IMAGES VIA HIGH-DIMENSIONAL INTERPRETABLE TRANSPORTATION MAPS

If $\boldsymbol{x}$ is an image with domain $\mathbb{R}^{d>>1}$, then any non-trivial transportation map in this space is likely to be hard to optimize for as well as uninterpretable. However, if P_{src}, P_{tgt} can be expressed on some *interpretable* lower dimensional manifold which is learned by some manifold-invertible function

$g : \mathbb{R}^d \to \mathbb{R}^k$ where $k < d$, we can project P_{src}, P_{tgt} onto this latent space and solve for an interpretable mapping such that it aligns the distributions in the latent space, $P_{T(g(\boldsymbol{x}))} \approx P_{g(\boldsymbol{y})}$. Note, in practice, an encoder-decoder with an interpretable latent space can be used for g, however, requiring g to be exactly invertible allows for mathematical simplifications, which we will see later. For explainability purposes, we can use g^{-1} to re-project $T\left(g(\boldsymbol{x})\right)$ back to $\mathbb{R}^d$ in order to display the transported image to an operator. With this, we can define our set of high dimensional interpretable transport maps: $\Omega_{\text{high-dim}} :- \left\{T : T = g^{-1}\left(\tilde{T}\left(g(\boldsymbol{x})\right)\right), \tilde{T} \in \Omega^{(k)}, g \in \mathcal{I}\right\}$ where $\Omega^{(k)}$ is the set of k-interpretable mappings (e.g., k-sparse or k-cluster maps) and $\mathcal{I}$ is the set of invertible functions with a interpretable (i.e. semantically meaningful) latent space.

Looking at our interpretable transport problem:

$$\underset{T \in \Omega_{\text{high-dim}}}{\arg\min}\ \mathbb{E}_{P_{src}}\left[c(\boldsymbol{x}, T(\boldsymbol{x}))\right] + \lambda_{Fid}\phi(P_{T(\boldsymbol{x})}), P_{\boldsymbol{y}}) \tag{6}$$

Although our transport is now happening in a semantically meaningful space, our transportation cost is still happening in the original raw pixel space. This is undesirable since we want a transport cost which penalizes large semantic movements, even if the true change in the pixel space is small (e.g., a change from “dachshund” to “hot dog” would be a large semantic movement). Sean: I really like the above example, but I’m willing to change it to something more somber/common if you think that’s best (e.g., a change from “cat” to “dog”) We can take a similar approach as before and instead calculate our transportation cost in the g space. This logic can similarly be applied to our divergence function ϕ (especially if ϕ is the Wasserstein distance, since this term can be seen as the residual shift not explained by T). Thus, calculating our cost and alignment functions within the latent space gives us:

$$\underset{g \in \mathcal{I}, \tilde{T} \in \Omega^{(k)}}{\arg\min}\ \mathbb{E}_{P_{src}}\left[c\left(g(\boldsymbol{x}), \tilde{T}\left(g(\boldsymbol{x})\right)\right)\right] + \lambda\phi(P_{\tilde{T}(g(\boldsymbol{x}))}, P_{g(\boldsymbol{y})}) \tag{7}$$

This formulation has a critical problem however. Since we are jointly learning our representation g and our transport map T, a trivial solution for the above minimization is for g to map each point to an arbitrarily small space such that the distance between any two points $c(g(\boldsymbol{x}), g(\boldsymbol{y})) \approx 0$, thus giving us a near zero cost regardless of how “far” we move points. To avoid this, we can use predefined image representation function h, e.g., the latter layers in Inception V3, and calculate pseudo-distances between transported images in this space. Because h expects an image as an input, we can utilize the invertibility of g and perform our cost calculation as: $c\left(h(\boldsymbol{x}), h\left(g^{-1}\left(\tilde{T}\left(g(\boldsymbol{x})\right)\right)\right)\right)$, or more simply, $c_h\left(\boldsymbol{x}, T(\boldsymbol{x})\right)$, where $T = g^{-1}\left(\tilde{T}\left(g(\boldsymbol{x})\right)\right)$. Similar to the previous equation, we also apply this h pseudo-distance to our divergence function to get ϕ_h. With this approach, we can still use g to jointly learn a semantic representation which is specific to our source and target domains (unlike h which is trained on images in general, e.g., ImageNet) and an interpretable transport map $\tilde{T}$ within g’s latent space. This gives us:

$$\underset{g \in \mathcal{I}, T \in \Omega}{\arg\min}\ \mathbb{E}_{P_{src}}\left[c_h\left(\boldsymbol{x}, T(\boldsymbol{x})\right)\right] + \lambda\phi_h(P_{T(\boldsymbol{x})}, P_{\boldsymbol{y}}) \tag{8}$$

While the above equation is an ideal approach, it can be difficult to use standard gradient approaches to optimize over in practice due it being a joint optimization problem and any gradient information having to first pass through h which could be a large neural network. To simplify this, we can optimize $\tilde{T}$ and g separately. With this, we can first find a g which properly encodes our source and target distributions into a semantically meaningful latent space, and then find the best interpretable transport to align the distributions in the fixed latent space. The problem can be even further simplified by setting the pre-trained image representation function h to be equal to the pretrained g, since the disjoint learning of g and $\tilde{T}$ removes the shrinking cost problem. By setting $h := g$, we can see that $c\left(h(\boldsymbol{x}), h \circ g^{-1} \circ \tilde{T} \circ g(\boldsymbol{x})\right) = c\left(g(\boldsymbol{x}), \tilde{T} \circ g(\boldsymbol{x})\right) = c_g(\boldsymbol{x}, \tilde{T}(\boldsymbol{x}))$, which simplifies Eqn. 8 back to our interpretable transport problem, Eqn. 6, where g is treated as a pre-processing step on the input images:

$$\underset{T \in \Omega}{\arg\min}\ \mathbb{E}_{P_{src}}\left[c(g(\boldsymbol{x}), g\left(T(\boldsymbol{x})\right))\right] + \lambda\phi_g(P_{T(\boldsymbol{x})}), P_{\boldsymbol{y}}) \tag{9}$$

Another way to simplify Eqn. 8 is to relax the constraint that g is manifold-invertible and instead use a pseudo-invertible function such as an encoder g and decoder g^+ structure where g^+ is a pseudo-inverse to g such that $g^+(g(\boldsymbol{x})) \approx \boldsymbol{x}$. This gives us:

$$\begin{aligned}\underset{\tilde{T}\in\tilde{\Omega},g,g^+}{\arg\min}\ \mathbb{E}_{P_{src}}\left[c_h\left(\boldsymbol{x}, g^+(\tilde{T}(g(\boldsymbol{x})))\right)\right] &+ \lambda_{Fid}\,\phi_h(P_{g^+(\tilde{T}(g(\boldsymbol{x})))}, P_{\boldsymbol{y}}) \\ &+ \lambda_{Recon}\,\mathbb{E}_{\frac{1}{2}P_{src}+\frac{1}{2}P_{tgt}}\left[L\left(\boldsymbol{x}, g^+(\tilde{T}(g(\boldsymbol{x})))\right)\right]\end{aligned} \tag{10}$$

where $L(\boldsymbol{x}, \cdot)$ is some reconstructive-loss function.

D.1 EXPLAINING A COLORIZED-MNIST SHIFT VIA HIGH-DIMENSIONAL INTERPRETABLE TRANSPORT

In this section we present a preliminary experiment showing the validity of our framework for explaining high-dimensional shifts. The experiment consists of using k-cluster maps to explain a shift in a colorized-version of MNIST, where the source environment is yellow/light red digits with a light grayscale background color (i.e. light gray) and the target environment consists of darker red digits and/or a darker grayscale background colors. Like the lower dimensional experiments before, our goal is to test our method on a shift where the ground truth is known and thus the explanation can validated against. We follow the framework presented in Eqn. 9, where the fixed g is a semi-supervised VAE Siddharth et al. (2017) which is trained on a concatenation of P_{src} and P_{tgt}. Our results show that k-cluster transport can capture the shift and explain the shift, however, we suspect the given explanation is interpretable because the ground truth is already known. Our hope is that future work will improve upon this framework by better finding a latent space which is interpretable and disentangled, leading to better latent mappings, and thus better high-dimensional shift explanations.

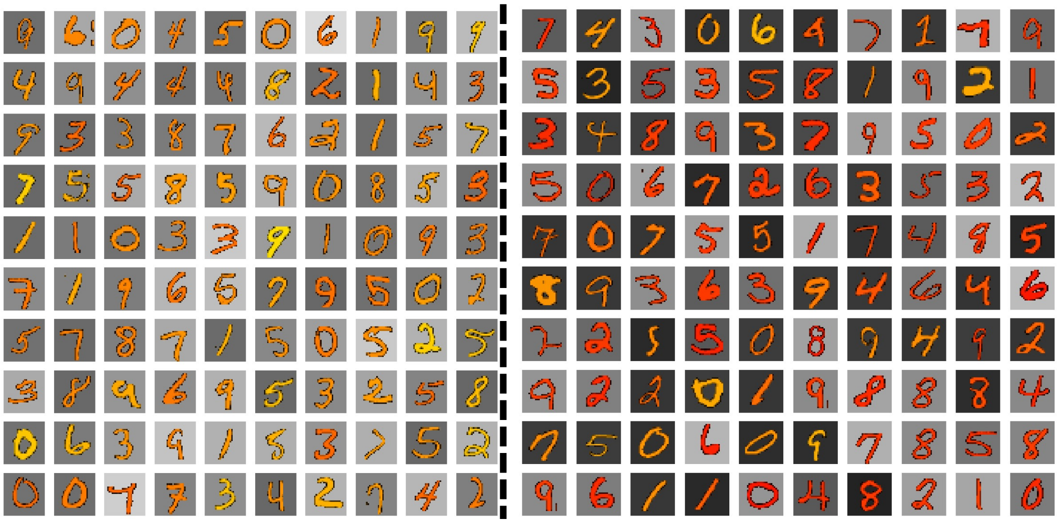


Figure 9: The left figure shows samples from the source environment which has lighter digits and backgrounds while the right figure shows the target environment which has darker digits and/or darker backgrounds

Data Generation The base data is the 60,000 grayscale handwritten digits from the MNIST dataset Deng (2012). We first colored each digit by copying itself along the red and green channel axes with an empty blue channel, yielding an initial dataset of yellow digits. We then randomly sampled 60,000 points from a two-dimensional Beta distribution with shape parameters, $\alpha = \beta = 5$. The first dimension of our Beta distribution represented how much of the green channel would be visible per sample meaning low values would result in a red image, while high values would result in a yellow image. The second dimension of our Beta distribution represented how white vs. black the background of the image would be, where $0 :=$ black background and $1 :=$ white background.

Specifically, the data was generated as follows. With $\boldsymbol{x}_{raw}$ representing a grayscale digit from the unprocessed MNIST dataset, a mask of representing the background was calculated $\mathbf{m} = \boldsymbol{x}_{raw} \leq 0.1$, where any pixel value below 0.1 is deemed to be the background (where each pixel value $\in [0, 1]$). Then, the foreground (i.e. digit) color was created $\boldsymbol{x}_{digit-color} = [(1-\mathbf{m}) \cdot \boldsymbol{x}_{raw},\ b_1 \cdot (1-\mathbf{m}) \cdot \boldsymbol{x}_{raw}, \mathbf{0}]$, where $\mathbf{0}$ is a zero-valued matrix matching the size of $\boldsymbol{x}_{raw}$ and $b_1 \sim \text{Beta}(\alpha, \beta)$. The background color was calculated via $\boldsymbol{x}_{back-color} = [b_2 \cdot \mathbf{m} \cdot \boldsymbol{x}_{raw},\ b_2 \cdot \mathbf{m} \cdot \boldsymbol{x}_{raw},\ b_2 \cdot \mathbf{m} \cdot \boldsymbol{x}_{raw}]$. Then $\boldsymbol{x}_{colored} = \boldsymbol{x}_{digit-color} + \boldsymbol{x}_{back-color}$, which results in a colorized MNIST digit with a stochastic foreground and background coloring.

The environments were created by setting the source environment to be any images where $b_1 \geq 0.4$ and $b_2 \geq 0.4$, i.e. any colorized digits that had over 40% of the green channel visible *and* a background at least 40% white, and the target environment is all other images. Informally, this split can be thought of as three sub-shifts: a shift which is only reddens the digit, a second shift which only a darkens the background, and a third shift which is both a digit reddening and background darkening. The environments can be seen in Fig. 9.

Model To encode and decode the colored images, we used a semi-supervised VAE (SSVAE) Siddharth et al. (2017). The SSVAE encoder consisted of an initial linear layer with input size of $3 \cdot 28 \cdot 28$ and a latent size of 1024. This was then multi-headed into a classification linear layer of size 1024 to 10, and for each sample with a label, digit label classification was performed on the output of this layer. The second head of the input layer was sent to a style linear layer of size 1024 to 50 which is to represent the style of the digit (and is not used in classification). The decoder followed a typical VAE decoder approach on a concatenation of the classification and style latent dimensions. The SSVAE was trained for 200 epochs on a concatenation of both P_{src} and P_{tgt} with 80% of the labels available per environment, and a batch size of 128 (for training details please see Siddharth et al. (2017)). The transport mapping was then found on the saved lower-dimensional embeddings.

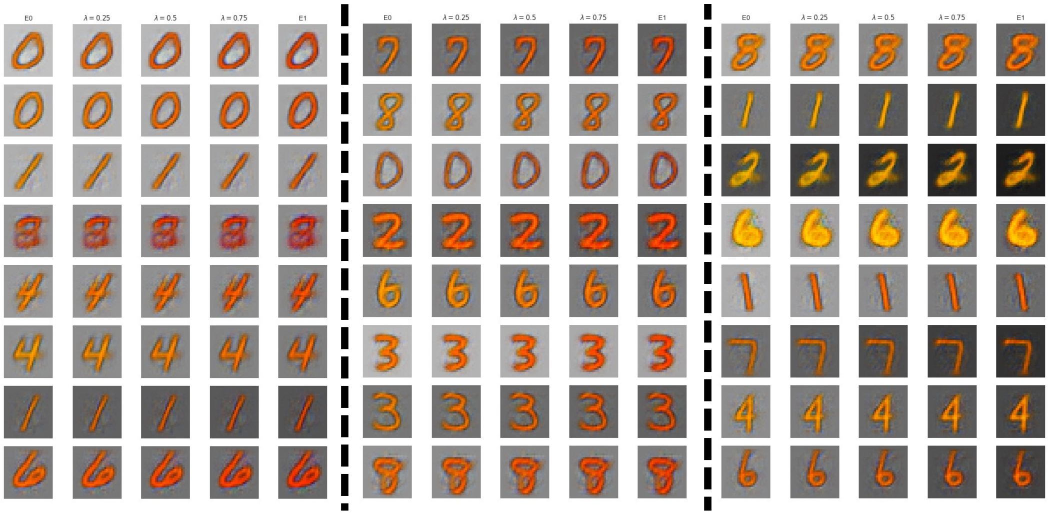


Figure 10: The linear interpolation explanations for the three clusters where the left cluster *seems* to explain the darkening digit shift, the right-most figure explains the shift which darkens the background, and the middle cluster explains the case where both digit and background darkens. For each cluster, the left-most digit $\boldsymbol{x}$ is the reconstruction of original encoding from the source distribution, the right-most digit is the cluster-based push-forward of that digit $T(\boldsymbol{x})$, and the three middle images are reconstructions of a linear interpolations $\lambda \cdot \boldsymbol{x} + (1-\lambda) \cdot T(\boldsymbol{x})$ with $\lambda \in \{0.25, 0.5, 0.75\}$.

Shift Explanation Results Given the shift is divided into three main sub-shifts, we used $k = 3$ cluster maps to explain the shift. We are followed the approach given in Eqn. 9, where the three cluster labels and transport were found in the 60 dimensional latent space using the algorithm given in Algorithm 1. Since our current approach is not able to find a latent space with disentangled and semantically meaningful axes, we cannot use the mean shift information per cluster as the explanation itself (as it is meaningless if the space is uninterpretable). Instead, we provide an operator with m samples from our source environment and the linear interpolation to the samples' push-forward versions under the target environment, for each cluster. The goal is for the operator to discern the meaning of each cluster's mean shift by finding the invariances across the m linear interpolations. The explanations can be seen in Fig. 10.

The linear interpolations from the first cluster (the left of Fig. 10) seem to show a darkening of the source digit, while keeping the background relatively constant. The third cluster (right-most side of the figure) represents the situation where only the background is darkened but the digit is not. Finally, the third cluster seems to explain the sub-shift where both the background and the digit are darkened. However, the changes made in the figures are quite faint, and without *a priori* knowledge of the shift it is possible that this could be an insufficient explanation. As mentioned in Section 6, this could be improved by finding a disentangled latent space with semantically meaningful dimensions, better approximating high dimensional empirical optimal transport maps, jointly finding a representation space and transport map like in Eqn. 4, and more; however, these advancements are out of scope for this work. We hope that this current preliminary experiment showcases the validity of using transportation maps to explain distribution shifts in images and inspires future work to build upon this foundation.