Identifiability Conditions for Domain Adaptation

Ishaan Gulrajani[1] **Tatsunori B. Hashimoto**[1]

Abstract

Domain adaptation algorithms and theory have relied upon an assumption that the observed data uniquely specify the correct correspondence between the domains. Unfortunately, it is unclear under what conditions this identifiability assumption holds, even when restricting ourselves to the case where a correct bijective map between domains exists. We study this bijective domain mapping problem and provide several new sufficient conditions for the identifiability of linear domain maps. As a consequence of our analysis, we show that weak constraints on the third moment tensor suffice for identifiability, prove identifiability for common latent variable models such as topic models, and give a computationally tractable method for generating certificates for the identifiability of linear maps. Inspired by our certification method, we derive a new objective function for domain mapping that explicitly accounts for uncertainty over maps arising from unidentifiability. We demonstrate that our objective leads to improvements in uncertainty quantification and model performance estimation.

1. Introduction

Given labeled data from a source domain and only unlabeled data from a different target domain, can we learn a classifier that performs well on the target domain? Many *unsupervised domain adaptation* algorithms have been proposed to learn such models, but the lack of labeled target-domain data makes it difficult to evaluate or trust the resulting models. Overcoming this challenge requires us to derive methods that provide guarantees on target-domain error in domain adaptation problems. More generally, these guarantees are a key step to answering the broader research question of "under what conditions are domain adaptation problems learnable?"

There is a long history of domain adaptation theory work that attempts to bound the target-domain error (e.g. Ben-David et al. (2010b) and survey in Redko et al. (2020)) but these methods either require that source and target distributions already be extremely similar (and often overlapping) or that the hypothesis class is highly constrained. While there are invariant representation approaches (Ganin et al., 2016) that avoid these drawbacks by learning representations that increase overlap between the domains, the target error guarantees for these models contain terms that require target-domain labels to estimate and can't easily be assumed away (Johansson et al., 2019).

There have been some recent successes in obtaining provable guarantees for complex predictors with little to no overlap in the setting where we assume the existence of a bijective *domain map* that relates the source and target inputs (Richardson & Weiss, 2021; Courty et al., 2016). In this setting, the problem of domain adaptation reduces to recovering the map. *Domain mapping* algorithms (e.g. Zhu et al., 2017) learn such a map by minimizing a distance between the distributions of source inputs and mapped target inputs over a family of maps with the true map achieving a distance of zero.

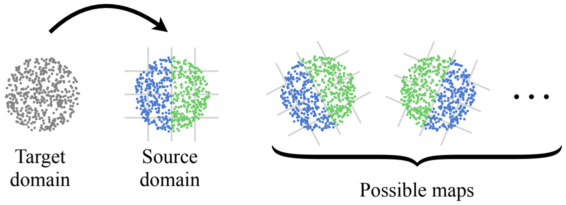

Figure 1. When mapping an unlabeled target domain to a labeled source domain, many possible maps can exist which yield different labelings.

Domain mapping suffers from a problem: besides the correct map, our map family may also contain may contain "spurious maps" which yield high target-domain error, but still align the input distributions. We have no way to disqualify these maps based on the data alone. We illustrate this problem in Figure 1: if both domains' inputs follow a uniform distribution over a circle, any rotation matrix will

[1] Stanford University, USA. Correspondence to: Ishaan Gulrajani <igul@stanford.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022. Copyright 2022 by the author(s).

align them, but the classifiers resulting from different rotations can be very different. In other words, the solution to the domain mapping problem can be *underspecified*.

Our ability to guarantee performance hinges on the *identifiability* of the map: among all the maps in the family, the correct map must be uniquely specified by the data. Despite the importance of this problem, provable identifiability conditions are not known for the basic case of a general linear map and the known results focus on the special cases of a ground-truth orthogonal linear map (Richardson & Weiss, 2021) and a positive definite linear map (Courty et al., 2016). Identifiability conditions for general linear maps would provide provable guarantees for a range of recent linear domain mapping methods in style transfer (Richardson & Weiss, 2021) and bilingual lexicon induction (Conneau et al., 2017; Zhang et al., 2017), and we focus on this goal for the majority of our work.

Figure 1 hints that a key roadblock to identifiability is the existence of symmetries in the distributions: the rotational symmetry of the circle prevents us from identifying a unique rotation that aligns the distributions. We develop this idea and show that linear symmetries can be ruled out through conditions on the third moment tensor, yielding sufficient conditions for the identifiability of a general linear map. Our conditions are broadly applicable: as one example, they imply identifiability for data generated by common latent variable models like topic models.

We complement our tensor-based identifiability conditions with a computationally tractable randomized algorithm that can provide high-probability guarantees for the identifiability of domain maps. As a proof of concept, we use this certification algorithm to prove the identifiability of linear maps over MNIST.

Finally, we develop new objective functions for the case where the domain map is not identifiable. Our objective attempts to estimate a model's *worst-case* error over a set of possible maps. This worst-case formulation can be used both as a training loss and a post-hoc evaluation, and we demonstrate that this approach provides improvements in model calibration and target-domain error estimates for neural-network based domain adaptation methods.

2. Problem Setup

In the *unsupervised domain adaptation* problem, we consider a source distribution $P^s(x, y)$ and a target distribution $P^t(x, y)$. We denote the corresponding random variables from these distributions as X^s, Y^s and X^t, Y^t. Given labeled source samples $(x_1^s, y_1^s) \dots (x_n^s, y_n^s) \sim P^s(x, y)$ and unlabeled target samples $x_1^t \dots x_m^t \sim P^t(x)$, our goal is to learn a predictor $h \in \mathcal{H}$ that achieves low target distribution loss

$$\mathcal{L}_T := \mathbb{E}[\ell(h(X^t), Y^t)],$$

where $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_{\geq 0}$ is a loss function. The fundamental challenge with the domain adaptation problem is that the target-domain conditional distribution $P^t(y|x)$ is unknown.

Our work considers *unsupervised domain mapping* methods as a way to address this problem. In many cases, there exists a bijective map that transforms target-domain inputs into corresponding source-domain inputs, and identifying this map allows us to solve the domain adaptation problem. Formally, we call a map T *admissible* if it aligns the marginal input distributions between domains as follows:

$$T(X^t) \overset{d}{=} X^s.$$

We call T *label-preserving* if it aligns the conditional distributions by ensuring that for all $x \in \mathcal{X}$,

$$Y|X^s = T(x) \quad \overset{d}{=} \quad Y|X^t = x.$$

Our goal then is to identify a map $T \in \mathcal{T}$ which is both admissible and label-preserving. If such an admissible and label-preserving map exists, we will call the domain mapping problem *realizable*, and we call a domain mapping problem *underspecified* if there exists at least one other *spurious* map which is admissible but not label preserving[1].

In the case that a domain mapping problem is realizable, the original domain adaptation problem is solvable by first learning a predictor h on our source domain as

$$h^s := \arg\min_{h \in \mathcal{H}} \mathbb{E}[\ell(h(X^s), Y^s)]$$

and then performing prediction via $h^t := h^s \circ T^*$, which is optimal on the class $\{h \circ T : h \in \mathcal{H}, t \in \mathcal{T}\}$. Assuming realizability, this argument reduces the problem of domain adaptation to identifying T^*, and we focus on this question of identifiability in the subsequent sections.

3. Identifiability Conditions for Domain Maps

Identifying an admissible label-preserving map is impossible in the presence of spurious maps. Recall our earlier example in Figure 1: identifying the label-preserving map was not possible due to the many spurious maps which exactly matched the marginal distribution of X^s. We will now derive conditions under which an admissible domain map ($T(X^t) \overset{d}{=} X^s$) is guaranteed to be unique. In this case, the

[1] This realizable map assumption always implies that the data is consistent with the *conditional shift* assumption from domain adaptation literature (Zhang et al., 2013). Often the distributions of X^s and X^t are disjoint, in which case it additionally implies that the data is consistent with the *covariate shift* assumption (Shimodaira, 2000).

only admissible map is the label-preserving one, and we can be assured that we have a solution to the domain adaptation problem.

Our analysis of domain map identifiability is based upon the study of symmetries: the map in Figure 1 was unidentifiable because of rotational symmetries in the distribution, and we expect complex real-world distributions without simple symmetries will have identifiable domain maps. We state this intuitive observation more formally in the following proposition.

Proposition 3.1. *For any pair of admissible linear maps T_1, T_2 from $P^t(x)$ to $P^s(x)$, there exists a linear symmetry S of $P^s(x)$ (that is, $SX^s \overset{d}{=} X^s$) such that $T_2 = ST_1$.*

Proof. Let $S = T_2 T_1^{-1}$, then $X^s \overset{d}{=} T_2 X^t \overset{d}{=} T_2 T_1^{-1} X^s$ so S is a linear symmetry of $P^s(x)$ satisfying $T_2 = ST_1$ which completes the proof. □

As a consequence, if the only linear symmetry for $P^s(x)$ is $S = I$ then admissible linear maps are identifiable, and if the only orthogonal symmetry is the identity map, then admissible orthogonal maps are identifiable.

The relationship between symmetries and identifiability implies that we can understand identifiability conditions through symmetry conditions, and we will provide a stratified analysis of linear symmetries in three different regimes of increasing generality: orthogonal linear symmetries using second moments, general linear symmetries using third moments, and general linear symmetries using sampled data.

3.1. Orthogonal linear identifiability

We begin with a simple setting where we restrict ourselves to orthogonal symmetries, yielding identifiability of orthogonal maps. In Figure 1, we saw that rotational and reflectional symmetries prevented us from identifying the map. These orthogonal symmetries can be ruled out on the basis of the eigenvectors and eigenvalues of the second moment matrix. Intuitively, uniqueness of the eigenvalues rules out rotational symmetries, and asymmetry of the marginal distributions rules out reflectional symmetries. This observation has been used in past work on linear orthogonal maps (Richardson & Weiss, 2021) but we state and prove this result formally to develop intuition for the more advanced identifiability results.

Proposition 3.2. *If all eigenvalues of the second moment of $P^s(x)$ are distinct, and the marginal distribution along each eigenvector v is asymmetric ($v^T X^s \overset{d}{\neq} -v^T X^s$), then $P^s(x)$ has no orthogonal symmetries other than the identity.*

Proof. Given in Appendix A.1. □

Identifiability among orthogonal maps is useful, but this still remains too restrictive to achieve our goal of obtaining general identifiability conditions for domain maps. To achieve this, we need conditions that rule out general linear symmetries.

We can reduce the problem of identifying a general linear symmetry to one of identifying an orthogonal symmetry by first whitening the domains, which ensures that any linear symmetries will become orthogonal ones. Unfortunately, whitening destroys precisely the information we previously relied upon to rule out orthogonal symmetries: the second moment of a whitened distribution is the identity, violating the assumptions of Proposition 3.2.

3.2. Linear identifiability with third moments

Although we can no longer rely on second moment information to rule out orthogonal symmetries, we find that there is an analogous condition to Proposition 3.2 that holds for third moment tensors. If the CP decomposition of the third moment tensor is unique and its rank is equal to the dimensionality of X^s, we can rule out any orthogonal symmetries even for whitened distributions.

Proposition 3.3. *Let X_1^s, X_2^s, X_3^s be three random variables from $P^s(x)$ and $M^s := \mathbb{E}[X_1^s \otimes X_2^s \otimes X_3^s]$ be its third moment tensor. Assume the following:*

1. *M^s has multilinear rank r.*
2. *M^s has a unique CP decomposition, up to the inherent ambiguities of CP decompositions.*
3. *There are no repeated weights in the CP decomposition of M^s when its factors are rescaled to have unit norm.*

Then the CP decomposition of M^s has at least r linearly independent mode-1 factors and for x in the span of these, $Sx = x$ for every orthogonal symmetry S. If $r = d$ then P^s has no orthogonal symmetries other than the identity.

Proof. Given in Appendix A.2. □

These conditions directly extend the existing second moment ones and allow us to certify the lack of linear symmetries by first whitening the distribution and verifying third moment conditions. This linear symmetry condition is sufficient to ensure identifiability of general linear maps (via Proposition 3.1) and provides a useful theoretical tool with which we can prove the uniqueness of domain maps for many probabilistic models with known third moments. For example, the extensive characterization of third moment tensors for latent variable models (Anandkumar et al., 2012; 2013) means that domain adaptation problems generated by many structured latent variable models are identifiable. We give one such example:

Remark 3.4. Let $P^s(x)$ be a whitened invertible linear transformation of the word vector distribution in the single topic model described by Anandkumar et al. (2012). Let X_1^s, X_2^s, X_3^s be sampled independently from the same document. By sec. 4.3.1 of Anandkumar et al. (2012), we have that M^s has an orthogonal decomposition whose factors are the k transformed topic vectors. If the model's topic weights are unique, then the assumptions of Proposition 3.3 are satisfied and it follows that for a vector x in the span of the transformed topic vectors, $Sx = x$ for every orthogonal symmetry S of $P^s(x)$.

More generally, we can use similar techniques to certify the uniqueness of maps on the basis of any odd moment tensor of order at least 3. The odd moments measure skewness (ruling out reflectional symmetries) and the singular values associated with the moment tensors can rule out rotational symmetries.

While Proposition 3.3 provides a clean theoretical characterization of identifiability for many probabilistic models, it is not a statistically or computationally efficient approach for providing identifiability guarantees. Statistically, third moment tensor decompositions can be sensitive to noise and require large sample sizes to be stable. Computationally, the CP decomposition is NP hard to compute and efficient approximations do not provide the types of guarantees needed for a certificate. We overcome these challenges by developing an alternative randomized algorithm that efficiently certifies the lack of linear symmetries without explicit computation of third moment conditions.

3.3. Efficiently certifying linear identifiability

We complement our third moment identifiability conditions (which are useful for characterizing identifiability of distributions with known third moments) with a computationally and statistically tractable identifiability certificate.

The key idea is that we can leverage any domain mapping algorithm $\textsc{Map}(\cdot,\cdot)$ as a way to rule out orthogonal symmetries on a whitened distribution $P^s(x)$. Our approach repeatedly applies a random orthogonal transformation R on data from $P^s(x)$ and runs $\textsc{Map}$ from un-transformed to transformed data. If there are no linear symmetries, the resulting domain maps will be approximately R, while any linear symmetry S will result in the mapping algorithm sometimes returning the map RS. We show that consistency over random transformations R provides a certificate for the lack of symmetry up to a small uncertainty set in operator norm.

Proposition 3.5. *Let* $\textsc{Map}(\cdot,\cdot)$ *be a stochastic mapping algorithm which takes two datasets and returns an orthogonal linear map. Let the random variable* $\mathbf{X} \in \mathbb{R}^{n\times d}$ *be a dataset drawn from* $P^s(x)$ *and let* $\mathbf{X}_1$ *be any fixed draw of* $\mathbf{X}$. *Let* R *be a random variable drawn from the Haar measure on the d-dimensional orthogonal group, and let* $T = R^{-1}\textsc{Map}(\mathbf{X}_1, \mathbf{X}R^T)$. *Let* $e_1, ..., e_n$ *be the canonical basis for* $\mathbb{R}^d$. *If* $\mathbb{P}(\|\mathrm{col}_i(T) - e_i\| \leq \epsilon) > 0.5$ *for all i, then for any orthogonal symmetry* S *of* $P^s(x)$, $\|S - I\|_{op} \leq 2\epsilon\sqrt{d}$.

Algorithm 1 CERTIFY

```
Requires: orthogonal mapping algorithm $\textsc{Map}(\cdot,\cdot)$.
Inputs: dataset $\mathbf{X}$ sampled from $P$, failure probability $\delta$, data dimension $d$.
Returns: bound on the operator norm of any orthogonal symmetry of $P$.
$k \leftarrow 1 + \log_2\left(\frac{1}{\delta}\right)$
Split $\mathbf{X}$ $k$ ways into $\mathbf{X}_1, \dots, \mathbf{X}_k$.
for $i = 2$ to $k$ do
    Sample $R \sim \textsc{Haar}[O(d)]$
    $T \leftarrow R^{-1}\textsc{Map}(\mathbf{X}_1, \mathbf{X}_i R^T)$
    $\epsilon_i \leftarrow \max_j \|\mathrm{col}_j(T) - e_j\|_2$
                                  {$e_j$ is the $j$th canonical basis vector}
end for
return $2\sqrt{d}\max_i \epsilon_i$
```

Proof. Given in Appendix A.3 □

Proposition 3.5 leads to CERTIFY, a computationally-tractable algorithm for certifying the lack of orthogonal symmetries, which we give in Algorithm 1. At a high level, CERTIFY splits a dataset, runs a mapping algorithm between random transformations of subsets of the dataset, computes a confidence bound for the value of ϵ in Proposition 3.5, and finally returns the corresponding operator-norm bound.

CERTIFY allows us to test whether any admissible map is unique by whitening P^s, running CERTIFY, and using Proposition 3.1 to turn the lack of symmetry into a lack of spurious domain maps. CERTIFY has runtime $\Theta(a \log \frac{1}{\delta})$ where a is the runtime of $\textsc{Map}$ (in our implementation, $\textsc{Map}$ is an SGD algorithm whose runtime is independent of δ).

Our results in this section demonstrate that applying random perturbations R, running $\textsc{Map}$, and measuring consistency is a useful framework to obtain computationally efficient certificates for identifiability. We now focus on how this insight can be used to develop new loss functions for domain mapping and adaptation.

4. A Worst-case Over Maps

Thus far we have focused our attention on certifying the performance of models by guaranteeing the identifiability of domain maps. However, domain maps used in practice are unlikely to be linear or exactly identifiable. In this case, we cannot identify a unique map, but we may be able to develop practical methods to quantify the degree of uncertainty in the domain map, and to provide worst-case bounds over all domain maps that match marginal distributions.

Our first observation is that small errors in identifiability can be acceptable as long as they are sufficiently small so as to not affect downstream prediction. Even in CERTIFY, the guarantee is that the map is identifiable up to a small error $2\epsilon\sqrt{d}$. Making this idea more formal, we can consider the set of all admissible translation maps between source and target and learn a predictor which minimizes the worst-case prediction error over these maps.

If all admissible maps lead to the same labeling, we can be confident in our predictions. On the other hand, if we find disagreements among admissible maps then we ought to be pessimistic about model performance. Formally, we will state this worst case approach as an upper bound on the target error of our model.

Proposition 4.1. *Let h be a target-domain predictor, h_s be a source-domain predictor, ℓ be a loss function satisfying the triangle inequality, and $\tilde{\mathcal{T}}$ be the set of all admissible maps. Then*

$$\mathcal{L}_T(h) \leq \mathcal{L}_S(h_s) + \sup_{T \in \tilde{\mathcal{T}}} \mathbb{E}_{P^t}\left[\ell(h(x), h_s(T(x)))\right].$$

Proof. Given in Appendix A.4. □

The intuition is that together with h_s, each possible map in $\tilde{\mathcal{T}}$ induces a labeling of the target domain, and knowing one of the labelings to be correct, we simply take the supremum over all possible such labelings. If our domain mapping problem is realizable, then we have a valid upper bound to the domain mapping performance of our predictor without relying on labeled target-domain data.

However, one drawback of this approach is that we cannot enumerate all admissible maps for a family of domain maps such as neural networks. We develop a heuristic approximation inspired by the randomized certificate. Instead of explicitly enumerating all maps, we consider a supremum over a small number of different maps $T_1, ..., T_n$ obtained by random restarts of a domain mapping algorithm. We call the resulting objective function Worst-case Over Maps *(WOMP)* and state it below:

$$\begin{aligned}&\mathcal{L}_{\text{WOMP}}(h, h_s, T_1, ..., T_n)\\ &:= \mathcal{L}_S(h_s) + \sup_{T \in \{T_1, ..., T_n\}} \mathbb{E}_{P^t}\left[\ell(h(x), h_s(T(x)))\right]\end{aligned}$$

Given a source-domain classifier and a small number of admissible maps from a mapping algorithm, the WOMP objective provides a straightforward estimate for the worst-case performance of a target-domain model. There are two main ways to use the WOMP objective. When computed on an already-trained model, it can be used as a proxy evaluation metric for the target loss. When used to train target-domain predictors by minimizing the WOMP objective, it yields predictors which attain tighter values of the bound and account for underspecification in their predictions. Evaluating the WOMP objective on a fixed set of m models requires time $\Theta(mn)$ where n is the dataset size.

Partial Maps and Invariant Representations: The WOMP objective is defined as a worst-case error over labelings induced by admissible maps. Since this definition only uses the maps to generate the predicted labels $h_s(T(x))$, it can be further generalized to even apply to *invariance* methods which do not define bijective maps between domains.

In invariant representation methods (Ganin et al., 2016; Tzeng et al., 2017), there are two encoders $\phi^s : \mathcal{X} \rightarrow \mathcal{Z}$ and $\phi^t : \mathcal{X} \rightarrow \mathcal{Z}$ and prediction is performed on a shared representation space $\mathcal{Z}$. While the success of such methods does not necessarily rely on the identifiability of any domain maps, recent work has observed that successful invariant representation learning is associated with invertibility of the encoders ϕ (Johansson et al., 2019; Zhao et al., 2019), in which case invariant representation methods behave as domain mapping methods and both methods have the same identifiability conditions.

Proposition 4.2. *Let $\phi^s, \phi^t \in \Phi$ be invertible encoders satisfying $\phi^s(X^s) \overset{d}{=} \phi^t(X^t)$. Let T^* be a label-preserving map which is identifiable among a family of maps $\mathcal{T}$. Assume $\forall \phi, \phi' \in \Phi . \phi \circ (\phi')^{-1} \in \mathcal{T}$. Then for all $x \in \mathcal{X}$,*

$$Y^s | \phi^s(X^s) = \phi^t(x) \quad \overset{d}{=} \quad Y | X^t = x.$$

Proof. Given in Appendix A.5. □

Because of this connection, we formulate WOMP in its most general form in our experiments, as the worst-case labeling over a set of admissible and potentially non-bijective maps.

5. Experiments

We present experiments validating our two algorithmic contributions, CERTIFY and WOMP. The focus of our work is conceptual rather than empirical; as such, our experiments consider illustrative simplified settings rather than realistic benchmarks.

5.1. CERTIFY

We use CERTIFY to certify the linear asymmetry of MNIST at a confidence level of 95%, which via Proposition 3.1 implies that domain mappings between transformations of MNIST are identifiable. We run two experiments: in the first experiment, we run CERTIFY on the pooled training and test

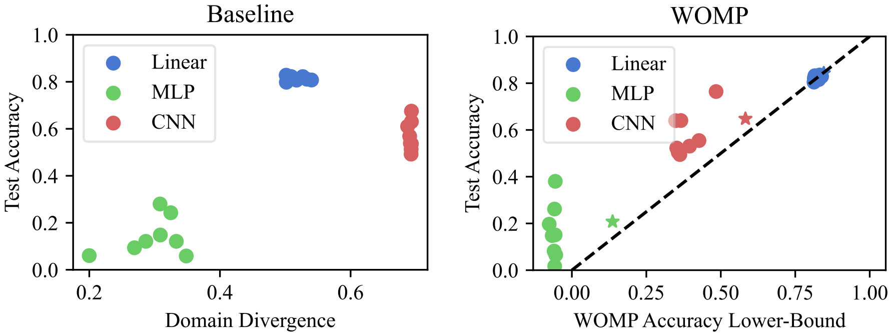

Figure 2. WOMP (right) provides useful estimates of a target-domain accuracy for ADDA models (circles), and even tighter estimates for WOMP-trained ones (stars). In contrast, the baseline (ADDA domain divergence, left) does not meaningfully correlate with target-domain accuracy.

splits of MNIST ($n = 70$K), and in the other, we generate $n = 1$M synthetic examples by applying the "InfiMNIST" data augmentation procedure of Loosli et al. (2007) and run CERTIFY on the augmented dataset.

As a preprocessing step, we apply PCA to whiten the data and truncate to the top 32 principal components, which explain 74% of the variance. By visual inspection, we confirm that the digit label remains easily identifiable. Recall that Proposition 3.5 makes no assumptions about the mapping algorithm (other than that it returns an orthogonal map) and therefore we are free to implement MAP within CERTIFY however we wish. We chose to instantiate MAP as an adversarial domain alignment algorithm with a number of choices to minimize optimization noise; we describe these, along with other experiment details, in Appendix B.1.

CERTIFY takes a dataset and a confidence level and outputs a bound on $\|S - I\|_{op}$ for any orthogonal symmetry S, which we translate into an ℓ_2 distance between the mappings of any unit-norm point under any pair of admissible maps. A small-enough distance means that all admissible maps are essentially the same. In particular, if the distance is smaller than a source-domain classifier's margin for a given point, then all admissible maps will yield the same label for that point. As a proxy for the margin, we compare our computed bounds to the median distance between an ℓ_2-normalized dataset example and its nearest neighbor in the dataset with a different label.

Table 1. CERTIFY results on MNIST (lower is better) at confidence level 0.95. We bound the ℓ_2 distance between admissible mappings of unit-normalized points. Given enough samples, our bound is smaller than the median distance to the nearest differently-labeled digit.

	ℓ_2 DIST.
CERTIFY BOUND, n=70K	0.56
CERTIFY BOUND, n=1M (DATA AUG.)	**0.16**
NEAREST NEIGHBOR WITH DIFF. LABEL	0.51

We present results in Table 1. Evaluated on the full MNIST dataset, CERTIFY certifies asymmetry up to an error which is slightly larger than the nearest-neighbor distance. However, we observe that this error is mostly an artifact of limited sample size. Given a larger sample size (through augmentation), CERTIFY successfully certifies that MNIST has essentially no linear symmetries, up to an error distance significantly less than the nearest-neighbor distance.

5.2. WOMP

We now demonstrate that accounting for the underspecification of domain adaptation problems via WOMP can be useful for estimating target-domain model performance, as well as improving uncertainty estimates.

Our first experiment considers the standard MNIST-USPS benchmark task (Long et al., 2013), in which MNIST digits form the source domain and USPS digits form the target domain. Our goal is to accurately estimate the performance of domain adaptation models across different function families and random seeds without relying on target-domain labels.

In the second task, we are interested in uncertainty quantification and calibration. We consider two datasets. The first is a semi-synthetic "Colored MNIST" dataset obtained by coloring half of the images in MNIST red and the other half green, yielding source and target domains. The second is a subset of the popular DomainNet benchmark (**?**), constructed by taking the "real photos" domain as the source and the "QuickDraw drawings" domain as the target. We filter the DomainNet data to the ten most frequent classes to construct a 10-way classification task. We train models with and without WOMP, and show that the WOMP objective leads to better calibration by accounting for unidentifiability

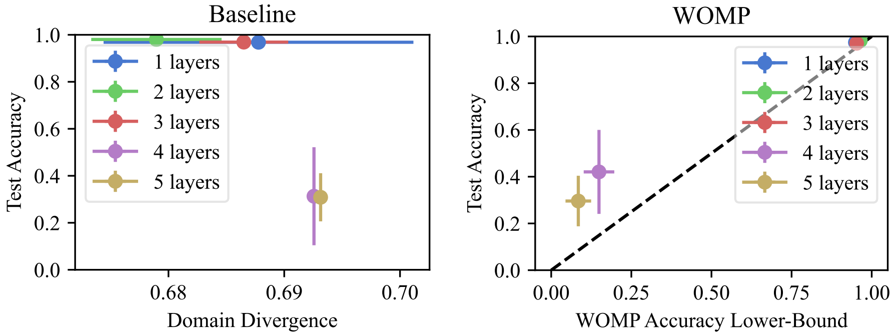

Figure 3. WOMP (right) can be used to select the number of CNN layers for ADDA models. The baseline hyperparameter selection metric (ADDA domain divergence, left) does not meaningfully separate high- and low-performing models. We plot means and standard deviations across 8 random seeds.

of the domain map.

ADDA Models Consistent with our findings in Section 5.1, we found in early experiments that domain mapping succeeds quite reliably on MNIST across a variety of map families. To demonstrate WOMP's benefits in a regime where underspecification exists, our experiments study non-invertible invariant representation models (see Section 4) rather than invertible domain maps. More specifically, we focus on ADDA models (Tzeng et al., 2017). An ADDA model consists of separate source and target encoders $\phi^s(x^s), \phi^t(x^t)$ which map inputs to a shared latent space, a classifier $f(z)$ defined on the latent space and trained using source-domain data, and a discriminator $D(z)$ used to align the distributions of $\phi^s(X^s)$ and $\phi^t(X^t)$. All three components are trained jointly in an adversarial game, and at test-time we classify target inputs via $h(x^t) = f(\phi^t(x^t))$.

Encoder Architectures The identifiability of an ADDA model depends on the function family of the encoders. Our experiments make use of three encoder architectures: orthogonal linear maps, MLPs, and CNNs. The MLPs have a single hidden layer of width 64. The CNNs have two layers of 5×5 kernels with stride 2 and widths 16 and 32, with no global pooling or fully-connected layers. These architectures and details were chosen to demonstrate WOMP's behavior across a range of underspecification levels; they are not intended to be state-of-the-art models for the benchmarks in question. In particular, the MLP encoders suffer heavily from underspecification, the linear maps suffer less, and the CNNs are essentially identifiable. For more details about architectures and hyperparameters see Appendix B.2.

5.2.1. Estimating Target-Domain Performance

Recall the problem from the introduction: without labeled target-domain data, we have no means of judging the success of our domain adaptation algorithm. Specifically, underspecification can cause existing domain adaptation methods to achieve nearly zero loss (even on held-out data) despite having high target-domain error. We run experiments on the MNIST-USPS task which show that by accounting for underspecification, the WOMP objective correlates with target-domain error across different model families, making it a useful quantity for model selection and evaluation.

We consider ADDA models with three choices of encoder architecture: orthogonal maps, CNNs, and MLPs. For each of these model families, we train models using ADDA over sixteen random seeds and compute the ADDA objective on held-out data for each seed.[2] Adversarial training in ADDA is known to be unstable (Creswell et al., 2018); to mitigate instability, we discard all but the eight best-ranking models (by ADDA objective) within each family.

Next, we compute the WOMP objective for each of the 27 models[3], using the eight models with the same architecture to generate the worst-case labelings for WOMP. For each encoder architecture, we additionally train a simple MLP target-domain classifier from scratch by minimizing the WOMP objective, again using the eight ADDA models of that architecture for WOMP's worst-case set.

We plot the WOMP objective against target-domain test accuracy in Figure 2 (right) and see that WOMP correctly bounds the test accuracy, and moreover ranks model families correctly without using any target-domain labels. Furthermore, we see that the WOMP-trained models (plotted

[2] ADDA is a min-max game whose objective includes an "adversarial domain divergence" term (Huang et al., 2017) which is not straightforward to measure, but we estimate it by training an independent domain discriminator after training the encoders, following Danihelka et al. (2017). See Appendix B.3 for more details about this procedure.

[3] Eight ADDA models with different random seeds plus one WOMP-trained model, for each of 3 architectures.

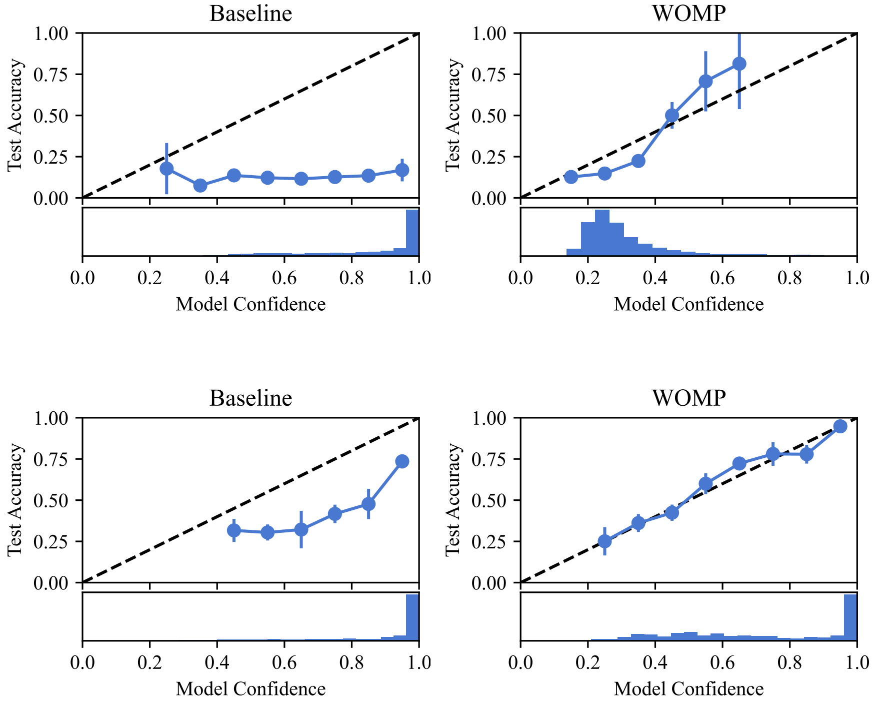

Figure 4. Reliability plots and confidence histograms for models trained with WOMP (right) and ADDA (left), on Colored MNIST (top) and a subset of DomainNet (bottom). The WOMP-trained models successfully make high- and low-confidence predictions, whereas the ADDA models are overconfident. We plot means and standard deviations across 5 random trials.

as stars) attain tighter values of the WOMP bound which correlate particularly well with their test accuracy. In contrast, Figure 2 (left) plots each ADDA model's test accuracy against the ADDA objective, evaluated on held-out data. We see that there is no correlation between the ADDA objective and the test performance across model families. These experiments suggest that by accounting for spurious maps, WOMP-style objectives may serve as a useful tool for model architecture selection in domain invariance methods.

Hyperparameter Selection Having shown that WOMP can be used to select between highly different model families (e.g. linear models and CNNs), we now show that it can also be used to select hyperparameters within a model family. Adopting the same setup as above, we train ADDA models on Colored MNIST with CNN encoders containing between 1 and 5 convolutional layers. Restricting the number of layers implicitly limits the CNN's receptive field, which eliminates spurious maps in this problem. We plot the results in Figure 3 and observe that WOMP successfully separates the high- and low-performing models, whereas the ADDA domain divergence does not.

5.2.2. Uncertainty Estimation

Because domain adaptation algorithms ignore underspecification, they can be highly confident about predictions which are ultimately incorrect. Unlike the i.i.d. model calibration setting (Gneiting et al., 2007; Guo et al., 2017a), this problem persists even in the infinite-data limit. To demonstrate this, we train domain adaptation models on Colored MNIST and a subset of DomainNet using ADDA with MLP encoders. After training, we calibrate each model's classifier using temperature scaling (Guo et al., 2017b) on held-out source-domain data. Figure 4 (left) shows a histograms of the models' confidence values as well as reliability plots (DeGroot & Fienberg, 1983; Niculescu-Mizil & Caruana, 2005), both computed on held-out target-domain data. The plot reveals that the models are overconfident in most of their predictions, and on Colored MNIST, that their confidence does not meaningfully correlate with predictive accuracy.

Next, we train MLP target-domain classifiers by minimizing the WOMP objective, using the eight ADDA MLP models as WOMP's worst-case set. We present the confidence histograms and reliability plots for these models in Figure 4 (right). The WOMP-trained models' confidence and predictive accuracy correlate clearly.

The WOMP-trained and ADDA-trained MLP models rely on the same encoder family and therefore both suffer from the same underspecification. Yet, Figure 4 shows that the WOMP-trained MLP model identifies a subset of examples which it can classify with greater confidence, and its predictions on this subset are indeed more accurate. To make these observations quantitative, we compute the Expected Calibration Error for the two models in Figure 4. We find that on Colored MNIST, the WOMP model attains an ECE of 0.10 ± 0.33 compared to 0.66 ± 0.03 for the baseline, and on DomainNet, WOMP attains 0.04 ± 0.01 compared to 0.24 ± 0.04 for the baseline.[4]

We observe consistent results for WOMP-trained models using linear and CNN ADDA encoders, which we give in Appendix C.2. For more experiment details, see Appendix B.

6. Related Work

Our work relates to domain adaptation theory and the study of symmetries. We discuss these connections below.

Domain Adaptation Theory and Methods In seminal work, Ben-David et al. (2010a) introduced the $\mathcal{H}\Delta\mathcal{H}$ divergence as a computable surrogate for the target-domain error. Unfortunately, $\mathcal{H}\Delta\mathcal{H}$ bounds contain 'joint train-test error' terms that cannot be measured without labeled test data and often return vacuous bounds for expressive models (See Redko et al. (2020) for a survey).

This second drawback has motivated invariant representation methods such as Ganin et al. (2016), which optimize $\mathcal{H}\Delta\mathcal{H}$ bounds using divergence-minimizing encoders. However, recent work has shown that these methods suffer errors due to unmeasurable joint-domain error terms and identifiability problems (Johansson et al., 2019; Zhao et al., 2019).

Our work is motivated by these observations of unidentifiability and complements existing domain adaptation theories such as $\mathcal{H}\Delta\mathcal{H}$ by providing conditions under which we can rule out identifiability concerns using unlabeled data alone.

Domain Mapping Domain mapping is an empirically successful family of methods that explicitly construct maps between source and target domains. This approach has been used in both visual domain adaptation (Zhu et al., 2017; Hoffman et al., 2018; Damodaran et al., 2018) and unpaired machine translation (Lample et al., 2017; Artetxe et al., 2017). Recent work has shown that even *linear* domain maps can be surprisingly effective in visual style transfer across domains (Richardson & Weiss, 2021) and word embedding alignment across languages (Conneau et al., 2017; Zhang et al., 2017).

The closest work to ours is Courty et al. (2016) who develop a method based on optimal transport with identifiability guarantees, but only for symmetric positive-definite linear maps. Our work greatly expands the set of known identifiability conditions (to general linear maps), provides a new computationally tractable certificate for identifiability, and develops algorithms that can improve uncertainty quantification when the ground truth map is not identifiable.

In complementary work, Zhang et al. (2013) and Gong et al. (2016) consider a more general family of domain adaptation problems in which the domain map can differ for each class. However, their identifiability conditions are restricted to a class of "location-scale transformations", whereas we show identifiability for the broader class of general linear maps.

Extrinsic Symmetries The identifiability of linear domain maps is equivalent to the existence of linear, extrinsic symmetries. This problem has been studied in the graphics literature, often in three dimensions (Mitra et al., 2013; Chertok & Keller, 2010; Ovsjanikov et al., 2011). While these existing results provide algorithms that can identify and return symmetries from shapes, they do not provide simple characterizations of identifiability (as in our third moment condition) or provide provable certificates that rule out symmetries (as in our randomized algorithm). Our work complements existing characterizations of symmetry, and our certificates for the lack of symmetries could be useful beyond domain adaptation problems.

7. Conclusions

We began with the question: "when does recovery of an admissible map guarantee success in domain adaptation?" Through conditions on the third moment tensor, we clarified that linear maps are often identifiable for parametric distributions like topic models and we derived CERTIFY, a computationally-tractable algorithm for certifying the identifiability of linear domain maps. We extended the ideas behind CERTIFY to develop WOMP, a method for estimating a bound on target-domain error even in nonlinear and unidentifiable problems, and validated our methods through simple illustrative experiments.

Our current work is limited to certifying linear maps and symmetries, but we believe that the same ideas and proof techniques extend directly to the nonlinear case by considering more general Lie group symmetries and their corresponding families of maps, providing a path towards certifying complex real world domain mapping methods.

Acknowledgements

We thank Martín Arjovsky, Kristy Choi, Yann Dubois, Leonidas Guibas, and Luke Metz for insightful comments.

[4] We report means and standard deviations across 5 random trials.

References

Anandkumar, A., Hsu, D., and Kakade, S. M. A method of moments for mixture models and hidden markov models. In *Conference on Learning Theory*, pp. 33–1. JMLR Workshop and Conference Proceedings, 2012.

Anandkumar, A., Ge, R., Hsu, D., Kakade, S. M., and Telgarsky, M. Tensor decompositions for learning latent variable models. *arXiv*, 2013.

Artetxe, M., Labaka, G., Agirre, E., and Cho, K. Unsupervised neural machine translation. *arXiv preprint arXiv:1710.11041*, 2017.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J. W. A theory of learning from different domains. *Machine Learning*, 79(1):151–175, 2010a.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J. W. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010b.

Chertok, M. and Keller, Y. Spectral symmetry analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32:1227–1238, 2010.

Conneau, A., Lample, G., Ranzato, M., Denoyer, L., and Jégou, H. Word translation without parallel data. *arXiv preprint arXiv:1710.04087*, 2017.

Courty, N., Flamary, R., Tuia, D., and Rakotomamonjy, A. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 39(9): 1853–1865, 2016.

Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., and Bharath, A. A. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 35(1):53–65, 2018.

Damodaran, B. B., Kellenberger, B., Flamary, R., Tuia, D., and Courty, N. Deepjdot: Deep joint distribution optimal transport for unsupervised domain adaptation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 447–463, 2018.

Danihelka, I., Lakshminarayanan, B., Uria, B., Wierstra, D., and Dayan, P. Comparison of maximum likelihood and gan-based training of real nvps. *arXiv preprint arXiv:1705.05263*, 2017.

DeGroot, M. H. and Fienberg, S. E. The comparison and evaluation of forecasters. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 32(1-2):12–22, 1983.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016.

Gneiting, T., Balabdaoui, F., and Raftery, A. E. Probabilistic forecasts, calibration and sharpness. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69 (2):243–268, 2007.

Gong, M., Zhang, K., Liu, T., Tao, D., Glymour, C., and Schölkopf, B. Domain adaptation with conditional transferable components. In *International conference on machine learning*, pp. 2839–2848. PMLR, 2016.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In *International Conference on Machine Learning (ICML)*, pp. 1321–1330, 2017a.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In *International Conference on Machine Learning*, pp. 1321–1330. PMLR, 2017b.

Hoffman, J., Tzeng, E., Park, T., Zhu, J.-Y., Isola, P., Saenko, K., Efros, A., and Darrell, T. Cycada: Cycle-consistent adversarial domain adaptation. In *International conference on machine learning*, pp. 1989–1998. PMLR, 2018.

Huang, G., Berard, H., Touati, A., Gidel, G., Vincent, P., and Lacoste-Julien, S. Parametric adversarial divergences are good task losses for generative modeling. *arXiv preprint arXiv:1708.02511*, 2017.

Johansson, F. D., Sontag, D., and Ranganath, R. Support and invertibility in domain-invariant representations. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 527–536. PMLR, 2019.

Lample, G., Conneau, A., Denoyer, L., and Ranzato, M. Unsupervised machine translation using monolingual corpora only. *arXiv preprint arXiv:1711.00043*, 2017.

Long, M., Wang, J., Ding, G., Sun, J., and Yu, P. S. Transfer feature learning with joint distribution adaptation. In *Proceedings of the IEEE international conference on computer vision*, pp. 2200–2207, 2013.

Loosli, G., Canu, S., and Bottou, L. Training invariant support vector machines using selective sampling. In Bottou, L., Chapelle, O., DeCoste, D., and Weston, J. (eds.), *Large Scale Kernel Machines*, pp. 301–320. MIT Press, Cambridge, MA., 2007. URL http://leon.bottou.org/papers/loosli-canu-bottou-2006.

Mescheder, L., Geiger, A., and Nowozin, S. Which training methods for gans do actually converge? In *International conference on machine learning*, pp. 3481–3490. PMLR, 2018.

Mitra, N. J., Pauly, M., Wand, M., and Ceylan, D. Symmetry in 3d geometry: Extraction and applications. *Computer Graphics Forum*, 32, 2013.

Niculescu-Mizil, A. and Caruana, R. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning*, pp. 625–632, 2005.

Ovsjanikov, M., Huang, Q., and Guibas, L. J. A condition number for non-rigid shape matching. *Computer Graphics Forum*, 30, 2011.

Redko, I., Morvant, E., Habrard, A., Sebban, M., and Bennani, Y. A survey on domain adaptation theory. *arXiv preprint arXiv:2004.11829*, 2020.

Richardson, E. and Weiss, Y. The surprising effectiveness of linear unsupervised image-to-image translation. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 7855–7861. IEEE, 2021.

Shimodaira, H. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of statistical planning and inference*, 90(2):227–244, 2000.

Tzeng, E., Hoffman, J., Saenko, K., and Darrell, T. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7167–7176, 2017.

Zhang, K., Schölkopf, B., Muandet, K., and Wang, Z. Domain adaptation under target and conditional shift. In *International conference on machine learning*, pp. 819–827. PMLR, 2013.

Zhang, M., Liu, Y., Luan, H., and Sun, M. Adversarial training for unsupervised bilingual lexicon induction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1959–1970, 2017.

Zhao, H., des Combes, R. T., Zhang, K., and Gordon, G. J. On learning invariant representation for domain adaptation. In *International Conference on Machine Learning (ICML)*, 2019.

Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 2223–2232, 2017.

A. Proofs of Things

A.1. Proof of Proposition 3.2

Proof. Let $\Sigma := \mathbb{E}[X^s(X^s)^T]$ be the second moment matrix of P, with eigenvalues $\lambda_1 < \ldots < \lambda_n$ and corresponding unit-norm eigenvectors $v_1, \ldots, v_n$. Let S be an orthogonal symmetry of $P^s(x)$ (that is, $SX^s \overset{d}{=} X^s$). Then for all $i \in [1, n]$ we have

$$\begin{aligned}
\Sigma v_i = \lambda_i v_i &\implies \mathbb{E}[X^s(X^s)^T]v_i = \lambda_i v_i \\
&\implies S\mathbb{E}[X^s(X^s)^T]v_i = \lambda_i S v_i \\
&\implies S\mathbb{E}[X^s(X^s)^T]S^T S v_i = \lambda_i S v_i \\
&\implies \mathbb{E}[(SX^s)(SX^s)^T]S v_i = \lambda_i S v_i \\
&\implies \Sigma S v_i = \lambda_i S v_i
\end{aligned}$$

where the last line is because S is a symmetry. Therefore $Sv_1, \ldots, Sv_n$ are eigenvectors of Σ with eigenvalues $\lambda_1, \ldots, \lambda_n$. Since the eigenvalues are distinct, each corresponding eigenvector is unique up to a sign ambiguity, and therefore for each i we have either $Sv_i = v_i$ or $Sv_i = -v_i$. We can use the asymmetric marginal to rule out the latter case as follows: for all i, we have

$$\begin{aligned}
(Sv_i)^T X^s &\overset{d}{=} (Sv_i)^T(SX^s) \\
&= v_i^T X^s \\
&\overset{d}{\neq} -v_i^T X^s
\end{aligned}$$

where the first line is because S is a symmetry, the second line is by orthogonality of S, and the third line is by asymmetry of $v^T X^s$. This implies that $Sv_i \neq -v_i$, so it must be that $Sv_i = v_i$ for all i, and therefore $S = I$. □

A.2. Proof of Proposition 3.3

Proof. Let $(w_i, a_i, b_i, c_i)_{i=1}^k$ be a CP decomposition of M^s (that is, $M^s = \sum_{i=1}^k w_i a_i \otimes b_i \otimes c_i$). Let $\|a_i\| = \|b_i\| = \|c_i\| = 1$ for all $i \in [1, k]$ and $w_i < w_{i+1}$ for all $i \in [1, k-1]$, where both of these conditions are without loss of generality since we can absorb the norms into w_i and permute the ordering i arbitrarily (recall that we've assumed all w_i to be distinct in this form). Let S be an orthogonal symmetry of $P^s(x)$ (that is, $SX^s \overset{d}{=} X^s$). Then we have

$$\begin{aligned}
M^s :=& \mathbb{E}[X_1^s \otimes X_2^s \otimes X_3^s] \\
=& \mathbb{E}[SX_1^s \otimes SX_2^s \otimes SX_3^s] \\
=& \mathbb{E}[X_1^s \otimes X_2^s \otimes X_3^s] \times_1 S \times_2 S \times_3 S \\
& \text{(since each transformation is linear along an appropriate matricization)} \\
=& \left(\sum_{i=1}^k w_i a_i \otimes b_i \otimes c_i\right) \times_1 S \times_2 S \times_3 S \\
=& \sum_{i=1}^k w_i (Sa_i) \otimes (Sb_i) \otimes (Sc_i).
\end{aligned}$$

Therefore, $(w_i, Sa_i, Sb_i, Sc_i)_{i=1}^k$ also form a CP decomposition of M^s. By assumption, the CP decomposition of M^s is unique up to rescaling of a_i, b_i, c_i and permutation of the ordering i, and we canonicalized these ambiguities in the beginning of the proof. Therefore the decomposition is completely unique and we have $a_i = Sa_i, b_i = Sb_i, c_i = Sc_i$.

Now, define $A := \begin{bmatrix} a_1 & \cdots & a_k \end{bmatrix}$ and it follows from the last paragraph that $A = SA$. We derive the rank of A as follows: any vector v in the left nullspace of A is a solution to $M^s \times_1 v = 0_{d \times d}$ and therefore also in the left nullspace of the mode-1 matricization of M^s. Therefore the rank of A is at least the rank of that matricization, which in turn is at least the multilinear rank of M, which by assumption is r.

Finally, since $A = SA$ we have $Sx = x$ for any $x \in \text{range}(A)$, and if $r = d$ then A is full-rank and therefore $S = I$, which concludes the proof.

□

A.3. Proof of Proposition 3.5

Proof. Let x be any point in $\mathbb{R}^d$, V be any subset of $\mathbb{R}^d$, and S be any orthogonal symmetry of $P^s(x)$ (that is, $X^s \overset{d}{=} SX^s$). First we will show that $\mathbb{P}(Tx \in V) = \mathbb{P}(Tx \in SV)$ where SV is the image of V under S. Define $R' = RS^{-1}$ and $T' = (R')^{-1}\textsc{Map}(\mathbf{X}_1, \mathbf{X}(R')^T)$. Then we have

$$\begin{aligned} T' &= (R')^{-1}\textsc{Map}(\mathbf{X}_1, \mathbf{X}(R')^T) \\ &= SR^{-1}\textsc{Map}(\mathbf{X}_1, \mathbf{X}(RS^{-1})^T) \\ &\overset{d}{=} SR^{-1}\textsc{Map}(\mathbf{X}_1, \mathbf{X}R^T) \\ &\overset{d}{=} ST \end{aligned}$$

where the third line is because S is a symmetry of P^s. By bijectivity of S we have $Tx \in V \iff STx \in SV$, and combining both statements gives $P(Tx \in V) = P(STx \in SV) = P(T'x \in SV)$.

Now, R and R' are identically distributed (because the Haar measure is invariant under multiplication by an orthogonal matrix), which implies that T and T' are identically distributed, which implies $P(T'x \in SV) = P(Tx \in SV)$. Combining both steps gives $P(Tx \in V) = P(Tx \in SV)$.

Next we show that if $P(Tx \in V) > 0.5$ then S must map V to a non-disjoint set. Assume that SV is disjoint from V, which leads to a contradiction:

$$\begin{aligned} P(Tx \in V \cup Tx \in SV) &= P(Tx \in V) + P(Tx \in SV) \\ &= 2P(Tx \in V) \\ &> 1. \end{aligned}$$

The next step is to show that for all i, $||Se_i - e_i|| \leq 2\epsilon$. For each i, let V_i be the sphere with radius ϵ centered at e_i. From the premise we have $\mathbb{P}(\|\text{col}_i(T) - e_i\| \leq \epsilon) > 0.5$ which is equivalent to $\mathbb{P}(Te_i \in V_i) > 0.5$, so from the previous step we have that SV_i must be non-disjoint from V_i. Since S is orthogonal, SV_i is also a sphere with center Se_i, and since SV_i and V_i are non-disjoint we have $||Se_i - e_i|| \leq 2\epsilon$.

Now, let z be any unit-norm vector in $\mathbb{R}^d$. Then we have

$$\begin{aligned} ||(S - I)z|| &= ||z_1(Se_1 - e_1) + ... + z_d(Se_d - e_d)|| \\ &\leq |z_1|||Se_1 - e_1|| + ... + |z_d|||Se_d - e_d|| \\ &\leq 2\epsilon(|z_1| + ... + |z_d|) \\ &\leq 2\epsilon\sqrt{d}. \end{aligned}$$

Finally, taking the supremum over z yields $\|S - I\|_{op} \leq 2\epsilon\sqrt{d}$ which is the claim. □

A.4. Proof of Proposition 4.1

Proof. Let $T^* \in \mathcal{T}$ be a label-preserving map. Then we have

$$\begin{aligned} \mathcal{L}_T(h) &= \mathbb{E}_{P^t}\left[\ell(h(x), y)\right] \\ &\leq \mathbb{E}_{P^t}\left[\ell(h(x), h_s(T^*(x)))\right] + \mathbb{E}_{P^t}\left[\ell(h_s(T^*(x)), y)\right] && \text{by the triangle inequality} \\ &= \mathbb{E}_{P^t}\left[\ell(h(x), h_s(T^*(x)))\right] + \mathcal{L}_S(h_s) && \text{by label-preservingness of } T^* \\ &\leq \sup_{T \in \tilde{\mathcal{T}}} \mathbb{E}_{P^t}\left[\ell(h(x), h_s(T(x)))\right] + \mathcal{L}_S(h_s) && \text{by the fact that } T^* \in \tilde{\mathcal{T}}, \end{aligned}$$

which is the claim (with the order of the terms switched). □

A.5. Proof of Proposition 4.2

Proof. An invertible invariant representation ϕ^s, ϕ^t implies an admissible domain map $\phi^t \circ (\phi^s)^{-1}$. The assumption implies that this map is in $\mathcal{T}$ and identifiability implies that it must be T^*. Then for all $x \in \mathcal{X}$ we have

$$\begin{aligned} Y^s|\phi^s(X^s) = \phi^t(x) &\overset{d}{=} Y^s|(\phi^t)^{-1}(\phi^s(X^s)) = x \\ &\overset{d}{=} Y^s|T^*(X^s) = x \\ &\overset{d}{=} Y^t|X^t = x. \end{aligned}$$

The second line is by invertibility of ϕ^s and the fourth line is by T^* being label-preserving. □

B. Experiment Details

B.1. CERTIFY

MAP is implemented as follows: we learn 128 orthogonal maps with standard adversarial training and different random seeds (the adversarial training details are the same as in Appendix B.2), take exponential moving averages of each map's parameters throughout training ($\epsilon = 0.999$), evaluate each by computing an adversarial divergence between the two splits (see Appendix B.3), take the top 12, average them together, and project the result onto the set of orthogonal matrices. CERTIFY splits the dataset and holds out the first split; we leverage this held-out split for tuning the hyperparameters of MAP.

For the MNIST experiment without data augmentation, we ℓ_2-regularize discriminator weights with $\lambda = 10^{-5}$, train for 8K steps, and use a discriminator gradient penalty weight of 1. For the experiment with augmentation, we use the same ℓ_2 regularization but train for 16K steps and use discriminator learning rate 10^{-4}.

B.2. ADDA training

All of the following hyperparameters were either tuned manually to minimize the adversarial divergence on a held-out set or set manually to conservative default values.

In all cases below, the discriminator is a ReLU MLP with 2 hidden layers of width 512. The classifier, encoders, and discriminator are trained jointly. The classifier minimizes a cross-entropy classification loss and the discriminator minimizes a binary cross-entropy with a gradient penalty regularizer as in Mescheder et al. (2018). The encoders minimize an equally-weighted sum of the negative discriminator loss and the classifier loss. All components are trained with Adam with $\beta_1 = 0.5, \beta_2 = 0.99$.

Classifier The classifier is an ReLU MLP with 1 hidden layer of width 128, trained jointly with the encoders and discriminator.

Linear encoder For Colored MNIST, inputs are PCA-whitened, truncated to 128 dimensions, and transformed by a random orthogonal matrix. For USPS-MNIST, inputs are PCA-truncated, but not whitened, and transformed by a random orthogonal matrix. Both source and target encoders are linear maps from 128-dimensional input space to a 64-dimensional latent space. Training proceeds for 40K steps with encoder learning rates 10^{-3}, classifier learning rate 10^{-3}, and discriminator learning rate 10^{-3}. We use a discriminator gradient penalty weight of 10. For MNIST-USPS experiments only, we use ℓ_2 regularization with $\lambda = 10^{-3}$ in the classifier and discriminator.

MLP encoder For Colored MNIST, inputs are PCA-whitened, truncated to 128 dimensions, and transformed by a random orthogonal matrix. For USPS-MNIST, inputs are PCA-truncated, but not whitened, and transformed by a random orthogonal matrix. We use MLPs with one hidden layer of width 64 and ReLU nonlinearity. The latent dimension is 64. Training proceeds for 20K steps with encoder learning rates 10^{-4}, classifier learning rate 10^{-4}, and discriminator learning rate 10^{-3}. We use a discriminator gradient penalty weight of 10. For MNIST-USPS experiments only, we use ℓ_2 regularization with $\lambda = 10^{-3}$ in the classifier and discriminator.

CNN encoder We use a 2-layer CNN with 5×5 kernels, stride 2, and ReLU nonlinearity. The channel widths are 16 and 32 for the first and second layers respectively. Training proceeds for 10K steps with encoder learning rates 10^{-5}, classifier learning rate 10^{-4}, and discriminator learning rate 10^{-4}. We use a discriminator gradient penalty weight of 1. For MNIST-USPS experiments only, we use ℓ_2 regularization with $\lambda = 10^{-2}$ in the classifier and $\lambda = 10^{-3}$ in the discriminator.

B.3. Adversarial divergence estimation

Given two datasets, we estimate the adversarial divergence between their distributions as follows: first, we split each dataset into 50% training, 25% validation, and 25% test splits. We train a binary classifier on the training set (MLP with 2 hidden layers of width 512), compute the validation loss every 10 steps during training, and take the test loss corresponding to the best validation loss. The final divergence is $\log 2$ minus the test loss.

The classifier is trained for 5000 steps with Adam ($\epsilon = 3 \times 10^{-4}$) and batch size 128. These hyperparameters were chosen by hand as conservative defaults.

B.4. WOMP evaluation and training

To mitigate the effects of instability in the adversarial training procedure, when evaluating the WOMP loss and training the WOMP predictor, we train 16 random restarts of the underlying ADDA model, compute adversarial divergences for each on held-out data (see Appendix B.3), and use only the top half of the models as inputs to the WOMP training procedure.

The WOMP-trained classifier is a 2-layer ReLU MLP with width 512, trained for 10K steps with Adam (default hyperparameters).

C. Additional Experiments

C.1. Estimating Target-Domain Performance on Colored MNIST

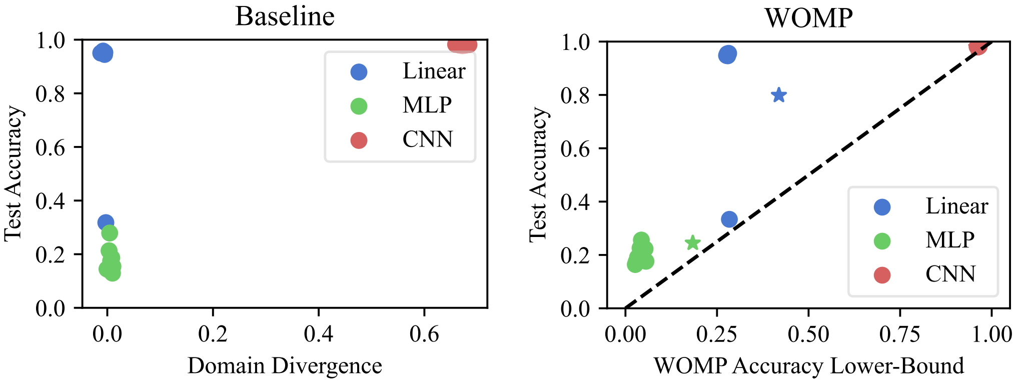

Figure 5. Model selection experiment results on Colored MNIST.

C.2. Uncertainty Estimation with Linear and CNN Encoders

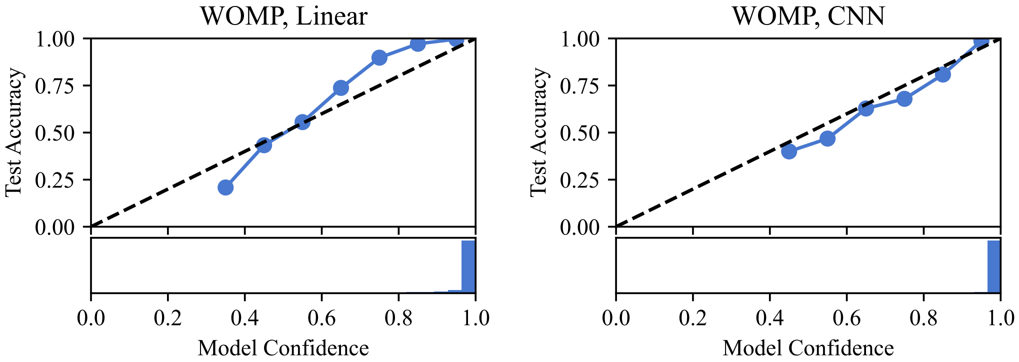

Figure 6. Uncertainty estimation results for WOMP with linear and CNN ADDA encoders.