

IN-N-OUT: PRE-TRAINING AND SELF-TRAINING USING AUXILIARY INFORMATION FOR OUT-OF-DISTRIBUTION ROBUSTNESS

Sang Michael Xie*, Ananya Kumar*, Robbie Jones*, Fereshte Khani, Tengyu Ma, Percy Liang
Stanford University
{xie, ananya, rmjones, fereshte, tengyuma, pliang}@cs.stanford.edu

ABSTRACT

Consider a prediction setting with few in-distribution labeled examples and many unlabeled examples both in- and out-of-distribution (OOD). The goal is to learn a model which performs well both in-distribution and OOD. In these settings, auxiliary information is often cheaply available for every input. How should we best leverage this auxiliary information for the prediction task? Empirically across three image and time-series datasets, and theoretically in a multi-task linear regression setting, we show that (i) using auxiliary information as input features improves in-distribution error but can hurt OOD error; but (ii) using auxiliary information as outputs of auxiliary pre-training tasks improves OOD error. To get the best of both worlds, we introduce In-N-Out, which first trains a model with auxiliary inputs and uses it to pseudolabel all the in-distribution inputs, then pre-trains a model on OOD auxiliary outputs and fine-tunes this model with the pseudolabels (self-training). We show both theoretically and empirically that In-N-Out outperforms auxiliary inputs or outputs alone on both in-distribution and OOD error.

1 INTRODUCTION

When models are tested on distributions that are different from the training distribution, they typically suffer large drops in performance (Blitzer and Pereira, 2007; Szegedy et al., 2014; Jia and Liang, 2017; AlBadawy et al., 2018; Hendrycks et al., 2019a). For example, in remote sensing, central tasks include predicting poverty, crop type, and land cover from satellite imagery for downstream humanitarian, policy, and environmental applications (Xie et al., 2016; Jean et al., 2016; Wang et al., 2020; Rußwurm et al., 2020). In some developing African countries, labels are scarce due to the lack of economic resources to deploy human workers to conduct expensive surveys (Jean et al., 2016). To make accurate predictions in these countries, we must extrapolate to out-of-distribution (OOD) examples across different geographic terrains and political borders.

We consider a semi-supervised setting with few in-distribution labeled examples and many unlabeled examples from both in- and out-of-distribution (e.g., global satellite imagery). While labels are scarce, auxiliary information is often cheaply available for every input and may provide some signal for the missing labels. Auxiliary information can come from additional data sources (e.g., climate data from other satellites) or derived from the original input (e.g., background or non-visible spectrum image channels). This auxiliary information is often discarded or not leveraged, and how to best use them is unclear. One way is to use them directly as input features (**aux-inputs**); another is to treat them as prediction outputs for an auxiliary task (**aux-outputs**) in pre-training. Which approach leads to better in-distribution or OOD performance?

Aux-inputs provide more features to potentially improve in-distribution performance, and one may hope that this also improves OOD performance. Indeed, previous results on standard datasets show that improvements in in-distribution accuracy correlate with improvements in OOD accuracy (Recht et al., 2019; Taori et al., 2020; Xie et al., 2020; Santurkar et al., 2020). However, in this paper we find that aux-inputs can introduce more spurious correlations with the labels: as a result, while aux-inputs often improve in-distribution accuracy, they can worsen OOD accuracy. We give examples of this trend on CelebA (Liu et al., 2015) and real-world satellite datasets in Sections 5.2 and 5.3.

Conversely, aux-output methods such as pre-training may improve OOD performance through auxiliary supervision (Caruana, 1997; Weiss et al., 2016; Hendrycks et al., 2019a). Hendrycks et al.

*Equal contribution.

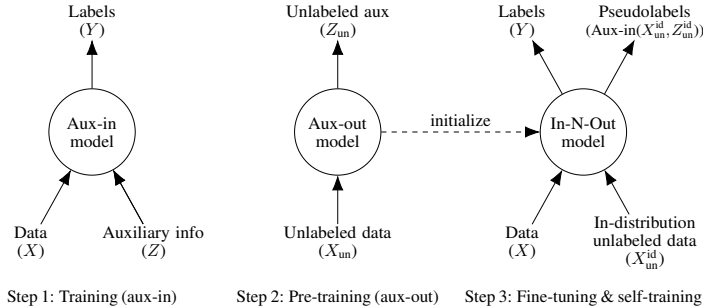


Figure 1: A sketch of the In-N-Out algorithm which consists of three steps: 1) use auxiliary information as input (Aux-in) to achieve good in-distribution performance, 2) use auxiliary information as output in pre-training (Aux-out), to improve OOD performance, 3) fine-tune the pretrained model from step 2 using the labeled data and in-distribution unlabeled data with pseudolabels generated from step 1 to improve in- and out-of-distribution.

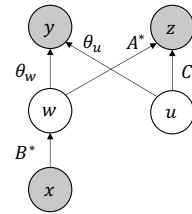


Figure 2: Graphical model for our theoretical setting: prediction task with input x , target y , and auxiliary information z , which is related to y through the latent variable w and latent noise u .

(2019a) show that pre-training on ImageNet can improve adversarial robustness, and Hendrycks et al. (2019b) show that auxiliary self-supervision tasks can improve robustness to synthetic corruptions. In this paper, we find that while aux-outputs improve OOD accuracy, the in-distribution accuracy is worse than with aux-inputs. Thus, we elucidate a tradeoff between in- and out-of-distribution accuracy that occurs when using auxiliary information as inputs or outputs.

To theoretically study how to best use auxiliary information, we extend the multi-task linear regression setting (Du et al., 2020; Tripuraneni et al., 2020) to allow for distribution shifts. We show that auxiliary information helps in-distribution error by providing useful features for predicting the target, but the relationship between the aux-inputs and the target can shift significantly OOD, worsening the OOD error. In contrast, the aux-outputs model first pre-trains on unlabeled data to learn a lower-dimensional representation and then solves the target task in the lower-dimensional space. We prove that the aux-outputs model improves robustness to *arbitrary* covariate shift compared to not using auxiliary information.

Can we do better than using auxiliary information as inputs or outputs alone? We answer affirmatively by proposing the In-N-Out algorithm to combine the benefits of auxiliary inputs and outputs (Figure 1). In-N-Out first uses an aux-inputs model, which has good in-distribution accuracy, to pseudolabel in-distribution unlabeled data. It then pre-trains a model using aux-outputs and finally fine-tunes this model on the larger training set consisting of labeled and pseudolabeled data. We prove that In-N-Out, which combines self-training and pre-training, further improves both in-distribution and OOD error over the aux-outputs model.

We show empirical results on CelebA and two remote sensing tasks (land cover and cropland prediction) that parallel the theory. On all datasets, In-N-Out improves OOD accuracy and has competitive or better in-distribution accuracy over aux-inputs or aux-outputs alone and improves 1–2% in-distribution, 2–3% OOD over not using auxiliary information on remote sensing tasks. Ablations of In-N-Out show that In-N-Out achieves similar improvements over pre-training or self-training alone (up to 5% in-distribution, 1–2% OOD on remote sensing tasks). We also find that using OOD (rather than in-distribution) unlabeled examples for pre-training is crucial for OOD improvements.

2 SETUP

Let $x \in \mathbb{R}^d$ be the input (e.g., a satellite image), $y \in \mathbb{R}$ be the target (e.g., crop type), and $z \in \mathbb{R}^T$ be the cheaply obtained auxiliary information either from additional sources (e.g., climate information) or derived from the original data (e.g., background).

Training data. Let P_{id} and P_{ood} denote the underlying distribution of (x, y, z) triples in-distribution and out-of-distribution, respectively. The training data consists of (i) in-distribution labeled data $\{(x_i, y_i, z_i)\}_{i=1}^n \sim P_{id}$, (ii) in-distribution unlabeled data $\{(x_i^{id}, z_i^{id})\}_{i=1}^{m_{id}} \sim P_{id}$, and (iii) out-of-distribution unlabeled data $\{(x_i^{ood}, z_i^{ood})\}_{i=1}^{m_{ood}} \sim P_{ood}$.

Goal and risk metrics. Our goal is to learn a model from input and auxiliary information to the target, $f : \mathbb{R}^d \times \mathbb{R}^T \rightarrow \mathbb{R}$. For a loss function ℓ , the in-distribution population risk of the model f is $R_{id}(f) = \mathbb{E}_{x, y, z \sim P_{id}}[\ell(f(x, z), y)]$, and its OOD population risk is $R_{ood}(f) = \mathbb{E}_{x, y, z \sim P_{ood}}[\ell(f(x, z), y)]$.

2.1 MODELS

We consider three common ways to use the auxiliary information (z) to learn a model.

Baseline. The baseline minimizes the empirical risk on labeled data while ignoring the auxiliary information (accomplished by setting z to 0):

$$\hat{f}_{\text{bs}} = \operatorname{argmin}_f \frac{1}{n} \sum_{i=1}^n \ell(f(x_i, 0), y_i). \quad (1)$$

Aux-inputs. The aux-inputs model minimizes the empirical risk on labeled data while using the auxiliary information as features:

$$\hat{f}_{\text{in}} = \operatorname{argmin}_f \frac{1}{n} \sum_{i=1}^n \ell(f(x_i, z_i), y_i). \quad (2)$$

Aux-outputs. The aux-outputs model leverages the auxiliary information z by using it as the prediction target of an auxiliary task, in hopes that there is a low-dimensional feature representation that is common to predicting both z and y . Training the aux-outputs model consists of two steps:

In the *pre-training* step, we use all the unlabeled data to learn a shared feature representation. Let $h: \mathbb{R}^d \rightarrow \mathbb{R}^k$ denote a feature map and $g_{z\text{-out}}: \mathbb{R}^k \rightarrow \mathbb{R}^T$ denote a mapping from feature representation to the auxiliary outputs. Let ℓ_{aux} denote the loss function for the auxiliary information. We define the empirical risk of h and $g_{z\text{-out}}$ as:

$$\hat{R}_{\text{pre}}(h, g_{z\text{-out}}) = \frac{1}{m_{\text{id}} + m_{\text{ood}}} \left(\sum_{i=1}^{m_{\text{id}}} \ell_{\text{aux}}(g_{z\text{-out}}(h(x_i^{\text{id}})), z_i^{\text{id}}) + \sum_{i=1}^{m_{\text{ood}}} \ell_{\text{aux}}(g_{z\text{-out}}(h(x_i^{\text{ood}})), z_i^{\text{ood}}) \right). \quad (3)$$

The estimate of the feature map is $\hat{h}_{\text{out}} = \operatorname{argmin}_h \min_{g_{z\text{-out}}} \hat{R}_{\text{pre}}(h, g_{z\text{-out}})$.

In the *transfer* step, the model uses the pre-trained feature map \hat{h}_{out} and the labeled data to learn the mapping $g_{y\text{-out}}: \mathbb{R}^k \rightarrow \mathbb{R}$ from feature representation to target y . We define the transfer empirical risk as:

$$\hat{R}_{\text{trans}}(\hat{h}_{\text{out}}, g_{y\text{-out}}) = \frac{1}{n} \sum_{i=1}^n \ell(g_{y\text{-out}}(\hat{h}_{\text{out}}(x_i)), y_i) \quad (4)$$

The estimate of the target mapping is $\hat{g}_{y\text{-out}} = \operatorname{argmin}_{g_{y\text{-out}}} \hat{R}_{\text{trans}}(\hat{h}_{\text{out}}, g_{y\text{-out}})$. The final aux-outputs model is

$$\hat{f}_{\text{out}}(x, z) = \hat{g}_{y\text{-out}}(\hat{h}_{\text{out}}(x)). \quad (5)$$

Like the baseline model, the aux-outputs model ignores the auxiliary information for prediction.

3 THEORETICAL ANALYSIS OF AUX-INPUTS AND AUX-OUTPUTS MODELS

We now analyze the baseline, aux-inputs, and aux-outputs models introduced in Section 2. Our setup extends a linear regression setting commonly used for analyzing multi-task problems (Du et al., 2020; Tripuraneni et al., 2020).

Setup. See Figure 2 for the graphical model. Let $w = B^*x \in \mathbb{R}^k$ be a low-dimensional latent feature ($k \leq d$) shared between auxiliary information z and the target y . Let $u \in \mathbb{R}^m$ denote unobserved latent variables not captured in x . We assume z and y are linear functions of u and w :

$$y = \theta_w^\top w + \theta_u^\top u + \epsilon, \quad (6)$$

$$z = A^*w + C^*u, \quad (7)$$

where $\epsilon \sim P_\epsilon$ denotes noise with mean 0 and variance σ^2 . As in Du et al. (2020), we assume the dimension of the auxiliary information T is greater than the feature dimension k , that is $T \geq k$, and that A^*, B^* and C^* have full rank (rank k). We also assume $T \geq m$, where m is the dimension of u .

Data. Let P_x and P_u denote the distribution of x and u in-distribution (ID), and let P'_x, P'_u denote the distribution x and u OOD. We assume x and u are independent, have distributions with bounded density everywhere, and have invertible covariance matrices. We assume the mean of u is zero in-

and out-of-distribution¹. We assume we have $n \geq m + d$ in-distribution labeled training examples and unlimited access to unlabeled data both ID and OOD, a common assumption in unsupervised domain adaptation theory (Sugiyama et al., 2007; Kumar et al., 2020; Raghunathan et al., 2020).

Loss metrics. We use the squared loss for the target and auxiliary losses: $\ell(\hat{y}, y) = (y - \hat{y})^2$ and $\ell_{\text{aux}}(z, z') = \|z - z'\|_2^2$.

Models. We assume all model families ($f, h, g_{z\text{-out}}, g_{y\text{-out}}$) in Section 2 are linear.

Let $\mathcal{S} = (A^*, B^*, C^*, \theta_w, \theta_u, P_x, P_u)$ denote a problem setting which satisfies all the above assumptions.

3.1 AUXILIARY INPUTS HELP IN-DISTRIBUTION, BUT CAN HURT OOD

We first show that the aux-inputs model (2) performs better than the baseline model (1) in-distribution. Intuitively, the target y depends on both the inputs x (through w) and latent variable u (Figure 2). The baseline model only uses x to predict y ; thus it cannot capture the variation in y due to u . On the other hand, the aux-inputs model uses x and z to predict y . Since z is a function of x (through w) and u , u can be recovered from x and z by inverting this relation. Note that u is unobserved but implicitly recovered. The aux-inputs model can then combine u and x to predict y better.

Let $\sigma_u^2 = \mathbb{E}_{u \sim P_u}[(\theta_u^\top u)^2]$ denote the (in-distribution) variance of y due to the latent variables u . The following proposition shows that if $\sigma_u^2 > 0$ then with enough training examples the aux-inputs model has lower in-distribution population risk than the baseline model.²

Proposition 1. *For all problem settings \mathcal{S} , P_ϵ , assuming regularity conditions (bounded x, u , sub-Gaussian noise ϵ , and $T = m$), and $\sigma_u^2 > 0$, for all $\delta > 0$, there exists N such that for $n \geq N$ number of training points, with probability at least $1 - \delta$ over the training examples, the aux-inputs model improves over the baseline:*

$$R_{id}(\hat{f}_{in}) < R_{id}(\hat{f}_{bs}). \quad (8)$$

Although using z as input leads to better in-distribution performance, we show that the aux-inputs model can perform worse than the baseline model OOD for any number of training examples. Intuitively, the aux-inputs model uses z , which can be unreliable OOD because z depends on u and u can shift OOD. In more detail, the aux-inputs model learns to predict $\hat{y} = \hat{\theta}_{x,in}^\top x + \hat{\theta}_{z,in}^\top z$, where the true output $y = \theta_x^\top x + \theta_z^\top z$, and $\hat{\theta}_{z,in}$ is an approximation to the true parameter θ_z , that has some error. Out-of-distribution u and hence z can have very high variance, which would magnify $(\hat{\theta}_{z,in} - \theta_z)^\top z$ and lead to bad predictions.

Example 1. *There exists a problem setting \mathcal{S} , P_ϵ , such that for every n , there is some test distribution P'_x, P'_u with:*

$$\mathbb{E}[R_{ood}(\hat{f}_{in})] > \mathbb{E}[R_{ood}(\hat{f}_{bs})] \quad (9)$$

3.2 PRE-TRAINING IMPROVES RISK UNDER ARBITRARY COVARIATE SHIFT

While using z as inputs (aux-inputs) can worsen performance relative to the baseline, our first main result is that the aux-outputs model (which pre-trains to predict z from x , and then transfers the learned representation to predict y from x) outperforms the baseline model for all test distributions.

Intuition. Referring to Figure 2, we see that the mapping from inputs x to auxiliary z passes through the lower dimensional features w . In the pre-training step, the aux-outputs model predicts z from x using a low rank linear model, and we show that this recovers the ‘bottleneck’ features w (up to symmetries; more formally we recover the rowspace of B^*). In the transfer step, the aux-outputs model learns a linear map from the lower-dimensional w to y , while the baseline predicts y directly from x . To warm up, *without distribution shift*, the expected excess risk only depends on the dimension of the input, and not the conditioning. That is, the expected excess risk in linear regression is exactly $d\sigma^2/n$, where d is the input dimension, so the aux-outputs trivially improves over the baseline since $\dim(w) < \dim(x)$. In contrast, the *worst case risk under distribution shift depends on the conditioning of the data*, which could be worse for w than x . Our proof shows that the worst case risk (over all x and u) is still better for the aux-outputs model because projecting to the low-dimensional feature representation ‘zeroes-out’ some error directions.

¹This is not limiting because bias in z can be folded into x .

²Since z is typically low-dimensional and x is high-dimensional (e.g., images), the aux-inputs model needs only a slightly larger number of examples before it outperforms the baseline.

Algorithm 1 In-N-Out

-
- Require:** in-distribution labeled data $\{(x_i, y_i, z_i)\}_{i=1}^n \sim P_{\text{id}}$,
in-distribution unlabeled data $\{(x_i^{\text{id}}, z_i^{\text{id}})\}_{i=1}^{m_{\text{id}}} \sim P_{\text{id}}$,
OOD unlabeled data $\{(x_i^{\text{ood}}, z_i^{\text{ood}})\}_{i=1}^{m_{\text{ood}}} \sim P_{\text{ood}}$
- 1: Learn $\hat{f}_{\text{in}} : (x, z) \mapsto y$ from in-distribution labeled data $\{(x_i, y_i, z_i)\}_{i=1}^n \sim P_{\text{id}}$
 - 2: Pre-train $g_{z\text{-out}} \circ \hat{h}_{\text{out}} : x \mapsto z$ on aux-outputs from all unlabeled data $\{(x_i^{\text{id}}, z_i^{\text{id}})\}_{i=1}^{m_{\text{id}}} \cup \{(x_i^{\text{ood}}, z_i^{\text{ood}})\}_{i=1}^{m_{\text{ood}}}$
 - 3: Return $\hat{f} = \hat{g} \circ \hat{h}_{\text{out}} : x \mapsto y$ trained on labeled and pseudolabeled data $\{(x_i, y_i)\}_{i=1}^n \cup \{(x_i^{\text{id}}, \hat{f}_{\text{in}}(x_i^{\text{id}}, z_i^{\text{id}}))\}_{i=1}^{m_{\text{id}}}$
-

Theorem 1. For all problem settings \mathcal{S} , noise distributions P_ϵ , test distributions P'_x, P'_u , and $n \geq m + d$ number of training points:

$$\mathbb{E}[R_{\text{ood}}(\hat{f}_{\text{out}})] \leq \mathbb{E}[R_{\text{ood}}(\hat{f}_{\text{bs}})]. \quad (10)$$

See Appendix A for the proof.

4 IN-N-OUT: COMBINING AUXILIARY INPUTS AND OUTPUTS

We propose the In-N-Out algorithm, which combines both the aux-inputs and aux-outputs models for further complementary gains (Figure 1). As a reminder: (i) The aux-inputs model ($x, z \rightarrow y$) is good in-distribution, but bad OOD because z can be misleading OOD. (ii) The aux-outputs model ($x \rightarrow y$) is better than the baseline OOD, but worse than aux-inputs in-distribution because it doesn't use z . (iii) We propose the In-N-Out model ($x \rightarrow y$), which uses pseudolabels from aux-inputs (stronger model) in-distribution to transfer in-distribution accuracy to the aux-outputs model. The In-N-Out model does not use z to make predictions since z can be misleading / spurious OOD.

In more detail, we use the aux-inputs model (which is good in-distribution) to pseudolabel in-distribution unlabeled data. The pseudolabeled data provides more effective training samples (self-training) to fine-tune an aux-outputs model pre-trained on predicting auxiliary information from all unlabeled data. We present the general In-N-Out algorithm in Algorithm 1 and analyze it in the linear multi-task regression setting of Section 2. The In-N-Out model $\hat{f} = \hat{g} \circ \hat{h}_{\text{out}}$ optimizes the empirical risk on labeled and pseudolabeled data:

$$\hat{g} = \underset{g}{\operatorname{argmin}} (1 - \lambda) \hat{R}_{\text{trans}}(\hat{h}_{\text{out}}, g) + \lambda \hat{R}_{\text{st}}(\hat{h}_{\text{out}}, \hat{f}_{\text{in}}, g) \quad (11)$$

where $\hat{R}_{\text{st}}(\hat{h}_{\text{out}}, \hat{f}_{\text{in}}, g) = \frac{1}{m_{\text{id}}} \sum_{i=1}^{m_{\text{id}}} \ell(g(\hat{h}_{\text{out}}(x_i^{\text{id}}), \hat{f}_{\text{in}}(x_i^{\text{id}}, z_i^{\text{id}})))$ is the loss of self-training on pseudolabels from the aux-inputs model, and $\lambda \in [0, 1]$ is a hyperparameter that trades off between labeled and pseudolabeled losses. In our experiments, we fine-tune \hat{g} and \hat{h}_{out} together.

Theoretical setup. Because fine-tuning is difficult to analyze theoretically, we analyze a slightly modified version of In-N-Out where we train an aux-inputs model to predict y given the features $\hat{h}_{\text{out}}(x)$ and auxiliary information z , so the aux-inputs model $\hat{g}_{\text{in}} : \mathbb{R}^k \times \mathbb{R}^T \rightarrow \mathbb{R}$ is given by $\hat{g}_{\text{in}} = \underset{g}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^n \ell(g(\hat{h}_{\text{out}}(x_i), z_i), y_i)$. The population self-training loss on pseudolabels from the aux-inputs model $\hat{g}_{\text{in}} \circ \hat{h}_{\text{out}}$ is: $R_{\text{st}}(\hat{h}_{\text{out}}, \hat{g}_{\text{in}}, g) = \mathbb{E}_{x, z \sim P_{\text{id}}} [\ell(g(\hat{h}_{\text{out}}(x)), \hat{g}_{\text{in}}(\hat{h}_{\text{out}}(x), z))]$, and we minimize the self-training loss: $\hat{g} = \underset{g}{\operatorname{argmin}} R_{\text{st}}(\hat{h}_{\text{out}}, \hat{g}_{\text{in}}, g)$. At test time given input x, z the In-N-Out model predicts $\hat{g}(\hat{h}_{\text{out}}(x))$. For the theory, we assume all models ($\hat{g}_{\text{in}}, \hat{g}$, and \hat{h}_{out}) are linear.

4.1 IN-N-OUT IMPROVES OVER PRE-TRAINING UNDER ARBITRARY COVARIATE SHIFT

We prove that In-N-Out helps on top of pre-training, as long as the auxiliary features give us information about y relative to the noise ϵ in-distribution—that is, if σ_u^2 is much larger than σ^2 .

To build intuition, first consider the special case where the noise $\sigma^2 = 0$ (equivalently, $\epsilon = 0$). Since u can be recovered from w and z , we can write y as a linear function of w and z : $y = \gamma_w^\top w + \gamma_z^\top z$. We train an aux-inputs model \hat{g}_{in} from w, z to y on finite labeled data. Since there is no noise, \hat{g}_{in} predicts y perfectly from w, z (we learn γ_w and γ_z). We use \hat{g}_{in} to pseudolabel a large amount of unlabeled data, and since \hat{g}_{in} predicts y perfectly from w, z , the pseudolabels are perfect. So here pseudolabeling gives us a much larger and correctly labeled dataset to train the In-N-Out model on.

The technical challenge is proving that self-training helps under arbitrary covariate shift even when the noise is non-zero ($\sigma^2 > 0$), so the aux-inputs model \hat{g}_{in} that we learn is accurate but not perfect.



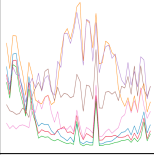
	CelebA	Cropland	Landcover
Visualization (x)			
Aux Info (z)	7 binary attributes	Vegetation, Lat/Lon	Meteorological Data
Target (y)	Male/female?	Cropland/not cropland?	Land cover class
ID-Split	People without hats	IA, MN, IL	Outside Africa
OOD-Split	People with hats	IN, KY	Africa

Figure 3: Summary of the datasets used in our experiments.

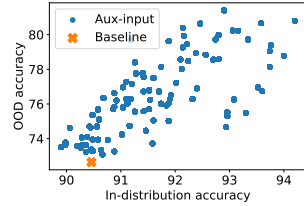


Figure 4: Correlation ($r = 0.72$) between in-distribution accuracy and OOD accuracy when adding 1 to 15 random auxiliary inputs in CelebA.

In this case, the pseudolabels have an error which propagates to the In-N-Out model self-trained on these pseudolabels, but we want to show that the error is lower than for the aux-outputs model. The error in linear regression is proportional to the noise of the target y , which for the aux-outputs model is $\sigma^2 + \sigma_u^2$. We show that the In-N-Out model uses the aux-inputs model to reduce the dependence on the noise σ_u^2 , because the aux-inputs model uses both w and z to predict y . The proof reduces to showing that the max singular value for the In-N-Out error matrix is less than the min-singular value of the aux-outputs error matrix with high probability. A core part of the argument is to lower bound the min-singular value of a random matrix (Lemma 3). This uses techniques from random matrix theory (see e.g., Chapter 2.7 in Tao (2012)); the high level idea is to show that with probability $1 - \delta$ each column of the random matrix has a (not too small) component orthogonal to all other columns.

Theorem 2. *In the linear setting, for all problem settings \mathcal{S} with $\sigma_u^2 > 0$, test distributions P'_x, P'_u , $n \geq m + d$ number of training points, and $\delta > 0$, there exists $a, b > 0$ such that for all noise distributions P_e , with probability at least $1 - \delta$ over the training examples and test example $x' \sim P'_x$, the ratio of the excess risks (for all σ^2 small enough that $a - b\sigma^2 > 0$) is:*

$$\frac{R_{in-out}^{ood} - R^*}{R_{out}^{ood} - R^*} \leq \frac{\sigma^2}{a - b\sigma^2} \tag{12}$$

Here $R^* = \min_{g^*, h^*} \mathbb{E}_{x', y', z' \sim P'} [\ell(g^*(h^*(x')), y')]$ is the min. possible (Bayes-optimal) OOD risk, $R_{in-out}^{ood} = \mathbb{E}_{y' \sim P'_{y'|x'}} [\ell(\hat{g}(\hat{h}_{out}(x')), y')]$ is the risk of the In-N-Out model on test example x' , and $R_{out}^{ood} = \mathbb{E}_{y' \sim P'_{y'|x'}} [\ell(\hat{g}_{y-out}(\hat{h}_{out}(x')), y')]$ is the risk of the aux-outputs model on test example x' . Note that R_{in-out}^{ood} and R_{out}^{ood} are random variables that depend on the test input x' and the training set X .

Remark 1. *As $\sigma \rightarrow 0$, the excess risk ratio of In-N-Out to Aux-outputs goes to 0, so the In-N-Out estimator is much better than the aux-outputs estimator.*

The proof of the result is in Appendix A.

5 EXPERIMENTS

We show on real-world datasets for land cover and cropland prediction that aux-inputs can hurt OOD performance, while aux-outputs improve OOD performance. In-N-Out improves OOD accuracy and has competitive or better in-distribution accuracy over other models on all datasets (Section 5.2). Secondly, we show that the tradeoff between in-distribution and OOD performance depends on the choice of auxiliary information on CelebA and cropland prediction (Section 5.3). Finally, we show that OOD unlabeled examples are important for improving OOD robustness (Section 5.4).

5.1 EXPERIMENTAL SETUP

We give a summary of considered datasets and setup here — see Figure 3 and Appendix B for details. Our datasets use auxiliary information both derived from the input (CelebA, Cropland) and from other sources (Landcover).

CelebA. In CelebA (Liu et al., 2015), the input x is a RGB image (resized to 64×64), the target y is a binary label for gender, and the auxiliary information z are 7 (of 40) binary-valued attributes derived from the input (e.g., presence of makeup, beard). We designate the set of images where the celebrity is wearing a hat as OOD. We use a ResNet18 as the backbone model architecture for all models (see Appendix B.1 for details).

	CelebA		Cropland		Landcover	
	ID Test Acc	OOD Acc	ID Test Acc	OOD Acc	ID Test Acc	OOD Test Acc
Baseline	90.46 ± 0.85	72.64 ± 1.39	94.50 ± 0.11	90.30 ± 0.75	75.92 ± 0.25	58.31 ± 1.87
Aux-inputs	92.36 ± 0.29	77.4 ± 1.33	95.34 ± 0.22	84.15 ± 4.23	76.58 ± 0.44	54.78 ± 2.01
Aux-outputs	94.0 ± 0.24	77.68 ± 0.59	95.12 ± 0.15	91.63 ± 0.21	72.48 ± 0.37	61.03 ± 0.97
In-N-Out (no pretrain)	93.8 ± 0.56	78.54 ± 1.31	94.93 ± 0.15	91.23 ± 0.61	76.54 ± 0.23	59.19 ± 0.98
In-N-Out	93.42 ± 0.36	79.42 ± 0.70	95.45 ± 0.16	91.94 ± 0.57	77.43 ± 0.39	61.53 ± 0.74
In-N-Out + repeated ST	93.76 ± 0.46	80.38 ± 0.68	95.53 ± 0.19	92.18 ± 0.40	77.10 ± 0.30	62.61 ± 0.58

Table 1: Accuracy (%) of various models using auxiliary information as input, output, or both. In-N-Out generally improves both in- and out-of-distribution over aux-inputs or aux-outputs alone. Results are averaged over 5 trials with 90% intervals. Repeated ST refers to one round of repeated self-training on top of In-N-Out.

Cropland. Crop type or cropland prediction is an important intermediate problem for crop yield prediction (Cai et al., 2018; Johnson et al., 2016; Kussul et al., 2017). The input x is a 50×50 RGB image taken by a satellite, the target y is a binary label that is 1 when the image contains majority cropland, and the auxiliary information z is the center location coordinate plus 50×50 vegetation-related bands. The vegetation bands in the auxiliary information z is derived from the original satellite image, which contains both RGB and other frequency bands. We use the Cropland dataset from Wang et al. (2020), with data from the US Midwest. We designate Iowa, Missouri, and Illinois as in-distribution and Indiana and Kentucky as OOD. Following Wang et al. (2020), we use a U-Net-based model (Ronneberger et al., 2015). See Appendix B.2 for details.

Landcover. Land cover prediction involves classifying the land cover type (e.g., “grasslands”) from satellite data at a location (Gislason et al., 2006; Rußwurm et al., 2020)). The input x is a time series measured by NASA’s MODIS satellite (Vermote, 2015), the target y is one of 6 land cover classes, and the auxiliary information z is climate data (e.g., temperature) from ERA5, a dataset computed from various satellites and weather station data (C3S, 2017). We designate non-African locations as in-distribution and Africa as OOD. We use a 1D-CNN to handle the temporal structure in the MODIS data. See Appendix B.3 for details.

Data splits. We first split off the OOD data, then split the rest into training, validation, and in-distribution test (see Appendix B for details). We use a portion of the training set and OOD set as in-distribution and OOD unlabeled data respectively. The rest of the OOD set is held out as test data. We run 5 trials, where we randomly re-generate the training/unlabeled split for each trial (keeping held-out splits fixed). We use a reduced number of labeled examples from each dataset (1%, 5%, 10% of labeled examples for CelebA, Cropland, and Landcover respectively), with the rest as unlabeled.

Repeated self-training. In our experiments, we also consider augmenting In-N-Out models with repeated self-training, which has fueled recent improvements in both domain adaptation and ImageNet classification (Shu et al., 2018; Xie et al., 2020). For one additional round of repeated self-training, we use the In-N-Out model to pseudolabel all unlabeled data (both ID and OOD) and also initialize the weights with the In-N-Out model. Each method is trained with early-stopping and hyperparameters are chosen using the validation set.

5.2 MAIN RESULTS

Table 1 compares the in-distribution (ID) and OOD accuracy of different methods. In all datasets, pre-training with aux-outputs improves OOD performance over the baseline, and In-N-Out (with or without repeated ST) generally improves both in- and out-of-distribution performance over all other models.

CelebA. In CelebA, using auxiliary information either as aux-inputs or outputs improves both ID (2–4%) and OOD accuracy (5%). We hypothesize this is because the auxiliary information is quite robust. Figure 4 shows that there is a significant correlation ($r = 0.72$) between ID and OOD accuracy for 100 different sets of aux-inputs, supporting results on standard datasets (Recht et al., 2019; Xie et al., 2020; Santurkar et al., 2020). In-N-Out achieves the best OOD performance and comparable ID performance even though there is no tradeoff between ID and OOD accuracy.

Remote sensing. In the remote sensing datasets, aux-inputs can induce a tradeoff where increasing ID accuracy hurts OOD performance. In cropland prediction, even with a small geographic shift (US Midwest), the baseline model has a significant drop from ID to OOD accuracy (4%). The aux-inputs model improves ID accuracy almost 1% above the baseline but OOD accuracy drops 6%. In land cover prediction, using climate information as aux-inputs decreases OOD accuracy by over 4% compared to the baseline. The aux-outputs model improves OOD, but decreases ID accuracy by 3% over the baseline.

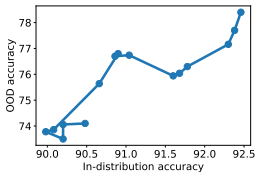


Figure 5: In-distribution vs. OOD accuracy on CelebA when sequentially adding a random set of 15 auxiliary inputs one-by-one. Even if adding all 15 auxiliary inputs improves both in-distribution and OOD accuracy, some intermediate in-distribution gains can hurt OOD.

	ID Test Acc	OOD Test Acc
Only in-distribution	69.73 \pm 0.51	57.73 \pm 1.58
Only OOD	69.92 \pm 0.41	59.28 \pm 1.01
Both	70.07 \pm 0.46	59.84 \pm 0.98

Table 2: Ablation study on the use of in-distribution vs. OOD unlabeled data in pre-training models on Landcover, where unlabeled sample size is standardized (much smaller than Table 1). Using OOD unlabeled examples are important for gains in OOD accuracy (%). Results are shown with 90% error intervals over 5 trials.

Improving in-distribution accuracy over aux-outputs. One of the main goals of the self-training step in In-N-Out is to improve the in-distribution performance of the aux-outputs model. We compare to oracle models that use a large amount of in-distribution labeled data to compare the gains from In-N-Out. In Landcover, the oracle model which uses 160k labeled ID examples gets 80.5% accuracy. In-N-Out uses 16k labeled examples and 150k unlabeled ID examples (with 50k unlabeled OOD examples) and improves the ID accuracy of aux-output from 72.5% to 77.4%, closing most (62%) of the gap. In Cropland, the oracle model achieves 95.6% accuracy. Here, In-N-Out closes 80% of the gap between aux-outputs and the oracle, improving ID accuracy from 95.1% to 95.5%.

Ablations with only pre-training or self-training. We analyze the individual contributions of self-training and pre-training in In-N-Out. On both cropland and land cover prediction, In-N-Out outperforms standard self-training on pseudolabels from the aux-inputs model (In-N-Out without pre-training), especially on OOD performance, where In-N-Out improves by about 1% and 2% respectively. Similarly, In-N-Out improves upon pre-training (aux-outputs model) both ID and OOD for both datasets.

5.3 CHOICE OF AUXILIARY INPUTS MATTERS

We find that the choice of auxiliary inputs affects the tradeoff between ID and OOD performance significantly, and thus is important to consider for problems with distribution shift. While Figure 4 shows that auxiliary inputs tend to simultaneously improve ID and OOD accuracy in CelebA, our theory suggests that in the worst case, there should be auxiliary inputs that worsen OOD accuracy. Indeed, Figure 5 shows that when taking a random set of 15 auxiliary inputs and adding them sequentially as auxiliary inputs, there are instances where an extra auxiliary input improves in-distribution but hurts OOD accuracy even if adding all 15 auxiliary inputs improves both ID and OOD accuracy. In cropland prediction, we compare using location coordinates and vegetation data as auxiliary inputs with only using vegetation data. The model with locations achieves the best ID performance, improving almost 1% in-distribution over the baseline with only RGB. Without locations (only vegetation data), the ID accuracy is similar to the baseline but the OOD accuracy improves by 1.5%. In this problem, location coordinates help with in-distribution interpolation, but the model fails to extrapolate to new locations.

5.4 OOD UNLABELED DATA IS IMPORTANT FOR PRE-TRAINING

We compare the role of in-distribution vs. OOD unlabeled data in pre-training. Table 2 shows the results of using only in-distribution vs. only OOD vs. a balanced mix of unlabeled examples for pre-training on the Landcover dataset, where unlabeled sample size is standardized across the models (by reducing to the size of the smallest set, resulting in 4x less unlabeled data). Using only in-distribution unlabeled examples does not improve OOD accuracy, while having only OOD unlabeled examples does well both in-distribution and OOD since it also has access to the labeled in-distribution data. For the same experiment in cropland prediction, the differences were not statistically significant, perhaps due to the smaller geographic shift (across states in cropland vs. continents in landcover).

6 RELATED WORK

Multi-task learning and weak supervision. Caruana and de Sa (2003) proposed using noisy features (aux-outputs) as a multi-task output, but do not theoretically analyze this approach. Wu et al. (2020) also study multi-task linear regression. However, their auxiliary tasks must have true parameters that are closely aligned (small cosine distance) to the target task. Similarly, weak supervision works assume access to weak labels correlated with the true label (Ratner et al., 2016; 2017). In our paper,

we make no assumptions about the alignment of the auxiliary and target tasks beyond a shared latent variable while also considering distribution shifts.

Transfer learning, pre-training, and self-supervision. We support empirical works that show the success of transfer learning and pre-training in vision and NLP (Krizhevsky et al., 2012; Simonyan and Zisserman, 2015; Devlin et al., 2019). Theoretically, Du et al. (2020); Tripuraneni et al. (2020) study pre-training in a similar linear regression setup. They show in-distribution generalization bound improvements, but do not consider OOD robustness or combining with auxiliary inputs. Hendrycks et al. (2019b) shows empirically that self-supervision can improve robustness to synthetic corruptions. We support these results by showing theoretical and empirical robustness benefits for pre-training on auxiliary information, which can be derived from the original input as in self-supervision.

Self-training for robustness. Raghunathan et al. (2020) analyze robust self-training (RST) (Carmon et al., 2019; Najafi et al., 2019; Uesato et al., 2019), which improves the tradeoff between standard and adversarially robust accuracy, in min-norm linear regression. Khani and Liang (2021) show how to use RST to make a model robust against a predefined spurious feature without losing accuracy. While related, we work in multi-task linear regression, study pre-training, and prove robustness to *arbitrary* covariate shifts. Kumar et al. (2020) show that repeated self-training on gradually shifting unlabeled data can enable adaptation over time. In-N-Out is complementary and may provide better pseudolabels in each step of this method. Chen et al. (2020) show that self-training can remove spurious features for Gaussian input features in linear models, whereas our results hold for general input distributions (with density). Zoph et al. (2020) show that self-training and pre-training combine for in-distribution gains. We provide theory to support this and also show benefits for OOD robustness.

Domain adaptation. Domain adaptation works account for covariate shift by using unlabeled data from a target domain to adapt the model (Blitzer and Pereira, 2007; Daumé III, 2007; Shu et al., 2018; Hoffman et al., 2018; Ganin et al., 2016). Often, modern domain adaptation methods (Shu et al., 2018; Hoffman et al., 2018) have a self-training or entropy minimization component that benefits from having a better model in the target domain to begin with. Similarly, domain adversarial methods (Ganin et al., 2016) rely on the inductive bias of the source-only model to correctly align the source and target distributions. In-N-Out may provide a better starting point for these domain adaptation methods.

7 DISCUSSION

Using spurious features for robustness. Counterintuitively, In-N-Out uses potentially spurious features (the auxiliary information, which helps in-distribution but hurts OOD accuracy) to improve OOD robustness. This is in contrast to works on removing spurious features from the model (Arjovsky et al., 2019; Ilyas et al., 2019; Chen et al., 2020). In-N-Out promotes utilizing all available information by leveraging spurious features as useful in-distribution prediction signals rather than throwing them away.

General robustness with unlabeled data. In-N-Out is an instantiation of a widely applicable paradigm for robustness: collect unlabeled data in all parts of the input space and learn better representations from the unlabeled data before training on labeled data. This paradigm has driven large progress in few-shot generalization in vision (Hendrycks et al., 2019a;b) and NLP (Devlin et al., 2019; Brown et al., 2020). In-N-Out enriches this paradigm by proposing that some features of the collected data can be used as input and output simultaneously, which results in robustness to arbitrary distribution shifts.

Leveraging metadata and unused features in applications. Many applications have inputs indexed by metadata such as location coordinates or timestamps (Christie et al., 2018; Yeh et al., 2020; Ni et al., 2019). We can use such metadata to join (in a database sense) other auxiliary data sources on this metadata for use in In-N-Out. This auxiliary information may often be overlooked or discarded, but In-N-Out provides a way to incorporate them to improve both in- and out-of-distribution accuracy.

Division between input features and auxiliary information. While a standard division between inputs and auxiliary information may exist in some domains, In-N-Out applies for any division of the input. An important further question is how to automatically choose this division under distribution shifts.

8 CONCLUSION

We show that while auxiliary information as inputs improve in-distribution and OOD on standard curated datasets, they can hurt OOD in real-world datasets. In contrast, we show that using auxiliary information as outputs by pretraining improves OOD performance. In-N-Out combines the strengths of auxiliary inputs and outputs for further improvements both in- and out-of-distribution.

9 ACKNOWLEDGEMENTS

We thank Sherrie Wang and Andreas Schlueter for their help in procuring remote sensing data, Daniel Levy for his insight in simplifying the proof of Theorem 1, Albert Gu for a key insight in proving Lemma 3 using tools from random matrix theory, as well as Shyamal Buch, Pang Wei Koh, Shiori Sagawa, and anonymous reviewers for their valuable help and comments. This work was supported by an Open Philanthropy Project Award, an NSF Frontier Award as part of the Center for Trustworthy Machine Learning (CTML). SMX was supported by an NDSEG Fellowship. AK was supported by a Stanford Graduate Fellowship. TM was partially supported by the Google Faculty Award, JD.com, Stanford Data Science Initiative, and the Stanford Artificial Intelligence Laboratory.

10 REPRODUCIBILITY

All code, data, and experiments are on CodaLab at [this link](#).

REFERENCES

- Sajjad Ahmad, Ajay Kalra, and Haroon Stephen. Estimating soil moisture using remote sensing data: A machine learning approach. *Advances in Water Resources*, 33(1):69–80, 2010.
- EA AlBadawy, A Saha, and MA Mazurowski. Deep learning for segmentation of brain tumors: Impact of cross-institutional training and testing. *Med Phys.*, 45, 2018.
- Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.
- John Blitzer and Fernando Pereira. Domain adaptation of natural language processing systems. *University of Pennsylvania*, 2007.
- Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.
- C3S. ERA5: Fifth generation of ECMWF atmospheric reanalyses of the global climate, 2017.
- Yaping Cai, Kaiyu Guan, Jian Peng, Shaowen Wang, Christopher Seifert, Brian Wardlow, and Zhan Li. A high-performance and in-season classification system of field-level crop types using time-series landsat data and a machine learning approach. *Remote Sensing of Environment*, 210:74–84, 2018.
- Yair Carmon, Aditi Raghunathan, Ludwig Schmidt, Percy Liang, and John C. Duchi. Unlabeled data improves adversarial robustness. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.
- Rich Caruana. Multitask learning. *Machine Learning*, 28:41–75, 1997.
- Rich Caruana and Virginia R. de Sa. Benefitting from the variables that variable selection discards. *Journal of Machine Learning Research (JMLR)*, 3, 2003.
- Yining Chen, Colin Wei, Ananya Kumar, and Tengyu Ma. Self-training avoids using spurious features under domain shift. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.
- Gordon Christie, Neil Fendley, James Wilson, and Ryan Mukherjee. Functional map of the world. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.
- Hal Daumé III. Frustratingly easy domain adaptation. In *Association for Computational Linguistics (ACL)*, 2007.
- R S DeFries and JRG Townshend. NDVI-derived land cover classifications at a global scale. *International Journal of Remote Sensing*, 15(17):3567–3586, 1994.
- Ruth DeFries, Matthew Hansen, and John Townshend. Global discrimination of land cover types from metrics derived from AVHRR pathfinder data. *Remote Sensing of Environment*, 54(3):209–222, 1995.

- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Association for Computational Linguistics (ACL)*, pages 4171–4186, 2019.
- Simon S. Du, Wei Hu, Sham M. Kakade, Jason D. Lee, and Qi Lei. Few-shot learning via learning the representation, provably. *arXiv*, 2020.
- Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, Francois Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research (JMLR)*, 17, 2016.
- Pall Oskar Gislason, Jon Atli Benediktsson, and Johannes R. Sveinsson. Random forests for land cover classification. *Pattern Recognition Letters*, 27(4):294–300, 2006.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Computer Vision and Pattern Recognition (CVPR)*, 2016.
- Dan Hendrycks, Kimin Lee, and Mantas Mazeika. Using pre-training can improve model robustness and uncertainty. In *International Conference on Machine Learning (ICML)*, 2019a.
- Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. Using self-supervised learning can improve model robustness and uncertainty. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019b.
- Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei A. Efros, and Trevor Darrell. Cycada: Cycle consistent adversarial domain adaptation. In *International Conference on Machine Learning (ICML)*, 2018.
- Daniel Hsu, Sham M. Kakade, and Tong Zhang. Random design analysis of ridge regression. In *Conference on Learning Theory (COLT)*, 2012.
- Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. *arXiv preprint arXiv:1905.02175*, 2019.
- Neal Jean, Marshall Burke, Michael Xie, W. Matthew Davis, David B. Lobell, and Stefano Ermon. Combining satellite imagery and machine learning to predict poverty. *Science*, 353, 2016.
- Robin Jia and Percy Liang. Adversarial examples for evaluating reading comprehension systems. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2017.
- Michael D. Johnson, William W. Hsieh, Alex J. Cannon, Andrew Davidson, and Frédéric Bédard. Crop yield forecasting on the canadian prairies by remotely sensed vegetation indices and machine learning methods. *Agricultural and Forest Meteorology*, 218:74–84, 2016.
- Fereshte Khani and Percy Liang. Removing spurious features can hurt accuracy and affect groups disproportionately. In *ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, 2021.
- Serkan Kiranyaz, Onur Avci, Osama Abdeljaber, Turker Ince, Moncef Gabbouj, and Daniel J Inman. 1d convolutional neural networks and applications: A survey. *arXiv preprint arXiv:1905.03554*, 2019.
- Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1097–1105, 2012.
- Ananya Kumar, Tengyu Ma, and Percy Liang. Understanding self-training for gradual domain adaptation. In *International Conference on Machine Learning (ICML)*, 2020.
- N. Kussul, M. Lavreniuk, S. Skakun, and A. Shelestov. Deep learning classification of land cover and crop types using remote sensing data. *IEEE Geoscience and Remote Sensing Letters*, 14(5): 778–782, 2017.
- David J. Lary, Amir H. Alavi, Amir H. Gandomi, and Annette L. Walker. Machine learning in geosciences and remote sensing. *Geoscience Frontiers*, 7(1):3–10, 2016.
- Ainong Li, Shunlin Liang, Angsheng Wang, and Jun Qin. Estimating crop yield from multi-temporal satellite data using multivariate regression and neural network techniques. *Photogrammetric Engineering & Remote Sensing*, 73(10):1149–1157, 2007.

- Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3730–3738, 2015.
- Ross Lunetta, Joseph F Knight, Jayantha Ediriwickremaand John G Lyon, and L Dorsey Worthy. Land-cover change detection using multi-temporal MODIS NDVI data. *Remote sensing of environment*, 105(2):142–154, 2006.
- Aaron E. Maxwell, Timothy A. Warner, and Fang Fang. Implementation of machine-learning classification in remote sensing: an applied review. *International Journal of Remote Sensing*, 39(9):2784–2817, 2018.
- Amir Najafi, Shin ichi Maeda, Masanori Koyama, and Takeru Miyato. Robustness to adversarial perturbations in learning from incomplete data. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.
- Jianmo Ni, Jiacheng Li, and Julian McAuley. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 188–197, 2019.
- Aditi Raghunathan, Sang Michael Xie, Fanny Yang, John C. Duchi, and Percy Liang. Understanding and mitigating the tradeoff between robustness and accuracy. In *International Conference on Machine Learning (ICML)*, 2020.
- Alexander Ratner, Stephen H Bach, Henry Ehrenberg, Jason Fries, Sen Wu, and Christopher Ré. Snorkel: Rapid training data creation with weak supervision. In *Very Large Data Bases (VLDB)*, number 3, pages 269–282, 2017.
- Alexander J Ratner, Christopher M De Sa, Sen Wu, Daniel Selsam, and Christopher Ré. Data programming: Creating large training sets, quickly. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 3567–3575, 2016.
- Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning (ICML)*, 2019.
- Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. *arXiv*, 2015.
- Marc Rußwurm, Sherrie Wang, Marco Korner, and David Lobell. Meta-learning for few-shot land cover classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 200–201, 2020.
- Shibani Santurkar, Dimitris Tsipras, and Aleksander Madry. Breeds: Benchmarks for subpopulation shift. *arXiv*, 2020.
- Rui Shu, Hung H. Bui, Hirokazu Narui, and Stefano Ermon. A DIRT-T approach to unsupervised domain adaptation. In *International Conference on Learning Representations (ICLR)*, 2018.
- K Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2015.
- Masashi Sugiyama, Matthias Krauledat, and Klaus-Robert Muller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research (JMLR)*, 8:985–1005, 2007.
- Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations (ICLR)*, 2014.
- Terrence Tao. *Topics in random matrix theory*. American Mathematical Society, 2012.
- Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, and Ludwig Schmidt. Measuring robustness to natural distribution shifts in image classification. *arXiv preprint arXiv:2007.00644*, 2020.
- Nilesh Tripuraneni, Michael I. Jordan, and Chi Jin. On the theory of transfer learning: The importance of task diversity. *arXiv*, 2020.

- Jonathan Uesato, Jean-Baptiste Alayrac, Po-Sen Huang, Robert Stanforth, Alhussein Fawzi, and Pushmeet Kohli. Are labels required for improving adversarial robustness? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.
- E. Vermote. MOD09A1 MODIS/terra surface reflectance 8-day L3 global 500m SIN grid V006. <https://doi.org/10.5067/MODIS/MOD09A1.006>, 2015.
- Sherrie Wang, William Chen, Sang Michael Xie, George Azzari, and David B. Lobell. Weakly supervised deep learning for segmentation of remote sensing imagery. *Remote Sensing*, 12, 2020.
- Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big Data*, 3, 2016.
- Sen Wu, Hongyang R. Zhang, and Christopher Ré. Understanding and improving information transfer in multi-task learning. In *International Conference on Learning Representations (ICLR)*, 2020.
- Michael Xie, Neal Jean, Marshall Burke, David Lobell, and Stefano Ermon. Transfer learning from deep features for remote sensing and poverty mapping. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2016.
- Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V. Le. Self-training with noisy student improves imagenet classification. *arXiv*, 2020.
- Christopher Yeh, Anthony Perez, Anne Driscoll, George Azzari, Zhongyi Tang, David Lobell, Stefano Ermon, and Marshall Burke. Using publicly available satellite imagery and deep learning to understand economic well-being in africa. *Nature Communications*, 11, 2020.
- Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin D. Cubuk, and Quoc V. Le. Rethinking pre-training and self-training. *arXiv*, 2020.