Calibrated ensembles can mitigate accuracy tradeoffs under distribution shift (Supplementary Material)

Ananya Kumar[1] **Tengyu Ma**[1] **Percy Liang**[1] **Aditi Raghunathan**[2]

[1]Computer Science Dept., Stanford University, Stanford, California, USA
[2]Computer Science Dept., Carnegie Mellon University, Pittsburgh, Pennsylvania, USA

Accepted for the 38th Conference on Uncertainty in Artificial Intelligence (UAI 2022).

1 PROOFS FOR SECTION 4

We begin with some standard background on Bayes optimal classifiers. When then prove the results in Section 4. By default, expectations are taken over all random variables.

1.1 BACKGROUND ON BAYES-OPTIMAL CLASSIFIERS

These results are all standard, but we include it as background information since different texts use different notations. Let $Z \in \mathcal{Z}$ denotes some features (that can be complicated functions of the input x, for example the output of a neural network), and let $Y \in \mathcal{Y}$ denote the label. Let P be a distribution over (Z, Y). The Bayes-optimal classifier predicts the most likely label y given features z.

Definition 1.1. *The Bayes-optimal classifier for P given features z is given by:*

$$y_*(z) = \arg\min_{y \in \mathcal{Y}} P(y \mid z). \tag{1.1}$$

The Bayes-optimal classifier has the minimum misclassification error of all possible classifiers that use $z \in \mathcal{Z}$ to predict $y \in \mathcal{Y}$. Formally, the error of a classifier $\widehat{y}$ is the probability that it gets the label incorrect.

Definition 1.2. *The error of a predictor $\widehat{y} : \mathcal{Z} \to \mathcal{Y}$ on distribution P is given by:*

$$\mathit{Err}_P(\widehat{y}) = P(Y \neq \widehat{y}(Z)), \tag{1.2}$$

Alternatively, we can look at the error for each Z, and then take the average over Z, which gives us:

Lemma 1.1. *The error of a predictor $\widehat{y} : \mathcal{Z} \to \mathcal{Y}$ on distribution P can be written as:*

$$\mathit{Err}_P(\widehat{y}) = \mathbb{E}[1 - P(Y = \widehat{y}(Z) \mid Z)]. \tag{1.3}$$

Proof. We can write the misclassification probability as an expectation over an indicator and then apply the law of total expectation.

$$\begin{aligned} P(Y \neq \widehat{y}(Z)) &= \mathbb{E}[\mathbb{I}(Y \neq \widehat{y}(Z))] && (1.4)\\ &= \mathbb{E}[\mathbb{E}[\mathbb{I}(Y \neq \widehat{y}(Z)) \mid Z]]. && (1.5)\end{aligned}$$

And then just write the inner expectation as a probability.

$$\begin{aligned} \mathbb{E}[\mathbb{E}[\mathbb{I}(Y \neq \widehat{y}(Z)) \mid Z]] &= \mathbb{E}[P(Y \neq \widehat{y}(Z) \mid Z)] && (1.6)\\ &= \mathbb{E}[1 - P(Y = \widehat{y}(Z) \mid Z)]. && (1.7)\end{aligned}$$

□

The Bayes-optimal classifier selects the y with the highest probability given z, so we have:

Lemma 1.2. *The error of the Bayes-optimal classifier y_* on a distributon P can be written as (where $Z \sim P$):*

$$\mathit{Err}_P(y_*) = \mathbb{E}[1 - \max_{y \in \mathcal{Y}} P(Y = y \mid Z)]. \tag{1.8}$$

Proof. The proof is immediate by substituting the definition of the Bayes-optimal classifier (Definition 1.1) into the alternative formula for the error in Lemma 1.1. □

From the above, it is clear that the Bayes-optimal classifier has lower error than any other classifier that uses only z, formalized below.

Lemma 1.3. *The bayes-optimal classifier (for P) has lower error than all classifiers $\widehat{y} : \mathcal{Z} \to \mathcal{Y}$:*

$$\mathit{Err}_P(y_*) \leq \mathit{Err}_P(\widehat{y}). \tag{1.9}$$

Proof. Beginning from Lemma 1.1, we have:

$$
\begin{aligned}
\mathrm{Err}_P(\widehat{y}) &= \mathbb{E}[1 - P(Y = \widehat{y}(Z) \mid Z)] && (1.10)\\
&\geq \mathbb{E}[1 - \max_{y \in \mathcal{Y}} P(Y = y \mid Z)] && (1.11)\\
&= \mathrm{Err}_P(y_*). && (1.12)
\end{aligned}
$$

□

As a simple corollary, we note that the accuracy of the Bayes-optimal classifier is at least the frequency of the most common label.

Corollary 1.1. *If y_* is bayes-optimal for P then,*

$$\mathit{Err}_P(y_*) \leq 1 - \max_{y \in \mathcal{Y}} P(Y = y) \tag{1.13}$$

So for example if P is balanced, then the Bayes-opt classifier will have accuracy at least $1/K$, where K is the number of classes.

Note that calibrated classifiers are Bayes-optimal given their outputs. Formally, let P be a distribution over (x, y), and suppose f is calibrated with respect to P. Let $z = f(x)$ and let P' be the induced distribution over (z, y). Then f is Bayes-optimal for P' given features z. The label distributions $P'(y)$ and $P(y)$ are the same, so Lemma 1.3 applies to any calibrated classifier.

1.2 PROOF OF PROPOSITION 4.1

Restatement of Proposition 4.1. *Suppose that f_{std} and f_{rob} are calibrated with respect to P_{id}, and that P_{id} is class-balanced. Let $h : \mathbb{R}^K \times \mathbb{R}^K \to \mathbb{R}^K$ be an arbitrary function that combines the standard and robust model's predictions, and let f_h be the resulting classifier: $f_h(x) = h(f_{\mathsf{std}}(x), f_{\mathsf{rob}}(x))$. The ensemble is better than any such combination classifier f_h: $\mathit{Err}_{\mathsf{id}}(f_{\mathsf{ens}}) \leq \mathit{Err}_{\mathsf{id}}(f_h)$.*

We first show that in the setting of the Proposition, we can write $P(y \mid f_{\mathsf{rob}}(x), f_{\mathsf{rob}}(x))$ in terms of $f_{\mathsf{rob}}(x)$ and $f_{\mathsf{std}}(x)$.

Lemma 1.4. *In the setting of Proposition 4.1, let $m \in \mathbb{R}^K$ be the log of the marginal probabilities $P(y)$:*

$$m_y = \log P(y), \quad \textit{for all } y \in [K]. \tag{1.14}$$

Then we have:

$$P(y \mid f_{\mathsf{std}}(x), f_{\mathsf{rob}}(x)) = \mathit{softmax}(f_{\mathsf{std}}(x) + f_{\mathsf{rob}}(x) - m)_y, \quad \textit{for all } y \in [K]. \tag{1.15}$$

In the balanced setting, where $P(y) = 1/K$ for all y, this simplifies to:

$$P(y \mid f_{\mathsf{std}}(x), f_{\mathsf{rob}}(x)) = \mathit{softmax}(f_{\mathsf{std}}(x) + f_{\mathsf{rob}}(x))_y, \quad \textit{for all } y \in [K]. \tag{1.16}$$

Proof. Fix $r = f_{\mathsf{rob}}(x)$ and $s = f_{\mathsf{std}}(x)$, where $r, s \in \mathbb{R}^K$. We first rewrite the probability of y given the robust and standard model outputs $P(y \mid r, s)$ in terms of the probability of y given each of the individual model outputs: $P(y \mid r)$ and $P(y \mid s)$. We do this for discrete random variables for simplicity, but the same result follows by using Bayes rule for general random variables.

$$
\begin{aligned}
P(y \mid r, s) &= \frac{P(r, s \mid y) P(y)}{P(r, s)} && \text{[Bayes rule]} && (1.17)\\
&= \frac{P(r \mid y) P(s \mid y) P(y)}{P(r, s)} && [r \perp s \mid y] && (1.18)\\
&= \frac{\left[\frac{P(y|r)P(r)}{P(y)} \frac{P(y|s)P(s)}{P(y)} P(y)\right.}{P(r, s)} && \text{[Bayes rule]} && (1.19)\\
&= \frac{P(y \mid r) P(y \mid s)}{P(y)} \left[\frac{P(r) P(s)}{P(r, s)}\right] && \text{[Algebra]} && (1.20)\\
& && && (1.21)
\end{aligned}
$$

Since r, s are fixed, we can denote the terms that do not depend on y by a constant c_1,

$$c_1 = \frac{P(r)P(s)}{P(r,s)}. \tag{1.22}$$

So then we can write:

$$P(y \mid r, s) = \frac{P(y \mid r)P(y \mid s)}{P(y)} c_1, \quad \text{for all } y \in [K]. \tag{1.23}$$

Now, we assumed $P(Y = y \mid r) = \text{softmax}(r)_y$ and $P(Y = y \mid s) = \text{softmax}(s)_y$ for all $y \in [K]$. For some constants $c_2, c_3 \in \mathbb{R}$, we can write this as: $P(Y = y \mid r) = \exp(r_y)/c_2$ and $P(Y = y \mid s) = \exp(s_y)/c_3$ for all $y \in [K]$. Substituting this into Equation 1.23, we get:

$$P(y \mid r, s) = \frac{\exp(r_y + s_y)}{P(y)} \frac{c_1}{c_2 c_3}, \quad \text{for all } y \in [K]. \tag{1.24}$$

Writing $1/P(y)$ as $\exp(-\log P(y))$, and setting $c_4 = \frac{c_1}{c_2 c_3}$, this gives us:

$$P(y \mid r, s) = c_4 \exp(r_y + s_y - \log P(y)), \quad \text{for all } y \in [K]. \tag{1.25}$$

Since the LHS is a probability, these must sum to 1 and so c_4 must be a normalizing constant, that is, $c_4 = 1/(\sum_{y \in [K]} \exp(r_y + s_y - \log P(y)))$. This gives us:

$$P(y \mid r, s) = \text{softmax}(r + s - m)_y, \quad \text{for all } y \in [K], \tag{1.26}$$

which is precisely Equation 1.15. In the balanced setting, we have $P(Y) = 1/K$ so we simply fold $P(Y)$ into the constant c_4, and get:

$$P(y \mid r, s) = \text{softmax}(r + s)_y, \quad \text{for all } y \in [K], \tag{1.27}$$

which is precisely Equation 1.16. □

Now we are ready to prove Proposition 4.1.

Proof of Proposition 4.1. We assumed the "balanced" setting where $P(y) = 1/K$ for all y. From Lemma 1.4, letting $f_{\text{ens}}(x) = f_{\text{std}}(x) + f_{\text{rob}}(x)$, we have:

$$P(y \mid f_{\text{std}}(x), f_{\text{rob}}(x)) = \text{softmax}(f_{\text{ens}}(x))_y, \tag{1.28}$$

So this means that the ensemble prediction is the Bayes optimal given $(f_{\text{std}}(x), f_{\text{rob}}(x))$:

$$\text{pred}(f_{\text{ens}}(x)) = \arg\max_y f_{\text{ens}}(x)_y = \arg\max_y \text{softmax}(f_{\text{ens}}(x))_y = \arg\max_y P(y \mid f_{\text{std}}(x), f_{\text{rob}}(x)). \tag{1.29}$$

But then from Lemma 1.3, any other predictor which uses only $(f_{\text{rob}}(x), f_{\text{std}}(x))$ must have higher error. This completes the proof.

Note that the inequality in the above proof is a strict inequality except in degenerate cases: as long as f_{std} and f_{rob} sometimes disagree in their predictions, and in some of these cases f_{std} assigns a higher probability to its predictions, and in some cases f_{rob} assigns a higher probability to its prediction, the inequalities will be strict inequalities. □

1.3 PROOF OF PROPOSITION 4.2

Restatement of Proposition 4.2. *If the OOD contains a mixture of suppressed features and missing spurious features i.e., $P_{\text{ood}} = \alpha P_\tau + (1 - \alpha) P_0$, and P_τ and P_0 are class-balanced, then we have $Err_{\text{ood}}(f_{\text{ens}}) \leq Err_{\text{ood}}(f_{\text{rob}})$ and $Err_{\text{ood}}(f_{\text{ens}}) \leq Err_{\text{ood}}(f_{\text{std}})$.*

Proof. We first note that errors are additive. That is, letting:

$$\text{Err}(P, f) = \mathbb{E}[\text{pred}(f(x)) \neq y], \text{ where } x, y \sim P, \tag{1.30}$$

we have:

$$\text{Err}(\alpha P_\tau + (1-\alpha)P_0, f) = \alpha \text{Err}(P_\tau, f) + (1-\alpha)\text{Err}(P_0, f) \tag{1.31}$$

So it suffices to prove that the ensemble is better than the standard and robust models for P_τ and P_0 separately.

Suppressed features. Let $\overline{f_{\mathsf{rob}}}(x) = \tau f_{\mathsf{rob}}(x)$ and $\overline{f_{\mathsf{std}}}(x) = \tau f_{\mathsf{std}}(x)$ be scaled versions of the standard and robust models. Definition 4.2 implies that $\overline{f_{\mathsf{rob}}}$ and $\overline{f_{\mathsf{std}}}$ are calibrated. Since we assumed P_τ is balanced, by Proposition 4.1, $\overline{f_{\mathsf{ens}}}$ given by $\overline{f_{\mathsf{ens}}}(x) = \tau f_{\mathsf{rob}}(x) + \tau f_{\mathsf{std}}(x)$ has optimal error on P_τ. But for all x, the predictions of f_{ens} and $\overline{f_{\mathsf{ens}}}$ are the same (multiplying the outputs of a model by a constant does not change the predicted output, which is the $\arg\max$). So f_{ens} also has optimal error on P_τ:

$$\text{Err}(P_\tau, f_{\mathsf{ens}}) \leq \text{Err}(P_\tau, f_{\mathsf{std}}), \text{ and } \text{Err}(P_\tau, f_{\mathsf{ens}}) \leq \text{Err}(P_\tau, f_{\mathsf{rob}}) \tag{1.32}$$

Note that these inequalities are strict inequalities except in degenerate cases: as long as f_{std} and f_{rob} sometimes disagree in their predictions, and in some of these cases f_{std} assigns a higher probability to its predictions, and in some cases f_{rob} assigns a higher probability to its prediction, the inequalities will be strict inequalities.

Missing spurious. If $f_{\mathsf{std}}(x) = 0$ almost surely, then $f_{\mathsf{ens}}(x) = f_{\mathsf{rob}}(x) + f_{\mathsf{std}}(x) = f_{\mathsf{rob}}(x)$ almost surely. Furthermore, if $f_{\mathsf{std}}(x) = 0$ then its error is lower bounded by $1 - \max_y P_0(y)$. On the other hand, $f_{\mathsf{rob}}(x)$ is calibrated and therefore Bayes-optimal given $z = f_{\mathsf{rob}}(x)$ so from Lemma 1.1 (e.g., see the the discussion below the Lemma for more details) has error at most $1 - \max_y P_0(y)$. So we have:

$$\text{Err}(P_0, f_{\mathsf{ens}}) = \text{Err}(P_0, f_{\mathsf{rob}}) \leq \text{Err}(P_0, f_{\mathsf{std}}) \tag{1.33}$$

Note that the inequality is a strict inequality except in a degenerate case (where the probability that f_{rob} predicts for the most common class $\arg\max_y P_0(y)$ is the same for all inputs). □

1.4 PROOF OF PROPOSITION 4.3

Restatement of Proposition 4.3. *If spurious features are anticorrelated OOD so that $P_{\mathsf{ood}} = P_{\mathsf{adv}}$, then even if P_{adv} is class-balanced, $Err_{\mathsf{ood}}(f_{\mathsf{rob}}) \leq Err_{\mathsf{ood}}(f_{\mathsf{ens}}) \leq Err_{\mathsf{ood}}(f_{\mathsf{std}})$.*

Proof. Let $X, Y \sim P_{\mathsf{ood}}$, and let $Z = (f_{\mathsf{std}}(X), f_{\mathsf{rob}}(X))$ be the predictions of the standard and robust models. Fix $z = (f_{\mathsf{std}}(x), f_{\mathsf{rob}}(x))$, and let $s = f_{\mathsf{std}}(x)$ and $r = f_{\mathsf{rob}}(x)$. We will analyze the errors for fixed $Z = z$ (showing that the robust model is better than the ensemble, which is better than the standard model). Since this is true for all z, we then use Lemma 1.1 (which is basically the law of total expectation), to get the desired result.

Bayes-opt classifier. Recall that for some $\alpha, \beta > 0$, we have $P_{\mathsf{adv}}(Y = y | f_{\mathsf{std}}(x)) = \text{softmax}(-\beta f_{\mathsf{std}}(x))_y$ for all x (note the minus sign), while $P_{\mathsf{adv}}(Y = y \mid f_{\mathsf{rob}}(x)) = \text{softmax}(\alpha f_{\mathsf{rob}}(x))_y$. Then, applying Lemma 1.4, we have:

$$P_{\mathsf{adv}}(y \mid (f_{\mathsf{std}}(x), f_{\mathsf{rob}}(x))) = \text{softmax}(\alpha f_{\mathsf{rob}}(x) - \beta f_{\mathsf{std}}(x))_y. \tag{1.34}$$

Rewriting this in terms of z, r, s, we have:

$$P_{\mathsf{adv}}(y \mid z) = \text{softmax}(\alpha r - \beta s)_y. \tag{1.35}$$

Ensemble vs. robust classifier. Let $j_{\mathsf{rob}} = \arg\max_y r_y$ be the robust model's prediction, and $j_{\mathsf{ens}} = \arg\max_y (r+s)_y$ be the ensemble model's prediction. Because j_{rob} is the $\arg\max$ of r, we have:

$$r_{j_{\mathsf{rob}}} \geq r_{j_{\mathsf{ens}}}. \tag{1.36}$$

Because j_{ens} is the $\arg\max$ of $r + s$, we have:

$$r_{j_{\mathsf{ens}}} + s_{j_{\mathsf{ens}}} \geq r_{j_{\mathsf{rob}}} + s_{j_{\mathsf{rob}}}. \tag{1.37}$$

Taking the negation of this, we get:

$$-r_{j_{\mathsf{rob}}} - s_{j_{\mathsf{rob}}} \geq -r_{j_{\mathsf{ens}}} - s_{j_{\mathsf{ens}}}. \tag{1.38}$$

Adding β times Inequality 1.38 to $(\alpha + \beta)$ times Inequality 1.36, we get:

$$\alpha r_{j_{\mathsf{rob}}} - \beta s_{j_{\mathsf{rob}}} \geq \alpha r_{j_{\mathsf{ens}}} - \beta s_{j_{\mathsf{ens}}}. \tag{1.39}$$

Since softmax is monotonic, we have:

$$\text{softmax}(\alpha r - \beta s)_{j_{\mathsf{rob}}} \geq \text{softmax}(\alpha r - \beta s)_{j_{\mathsf{ens}}}. \tag{1.40}$$

But from Equation 1.35 the LHS is the same as the robust model's probability of getting the label correct, and the RHS is the same as the ensemble's probability of getting the label correct:

$$P_{\mathsf{adv}}(Y = j_{\mathsf{rob}} \mid Z = z) \geq P_{\mathsf{adv}}(Y = j_{\mathsf{ens}} \mid Z = z). \tag{1.41}$$

Taking negations (to get the error), and then the expectation over $Z = z$, we get (note that below we write the error, which is why the sign is now flipped):

$$\text{Err}_{\mathsf{ood}}(f_{\mathsf{ens}}) \geq \text{Err}_{\mathsf{ood}}(f_{\mathsf{rob}}). \tag{1.42}$$

Which is what we wanted to show.

Ensemble vs. standard classifier. The argument is fairly analogous to the previous case, with some minor differences in the algebra in the first part. Let $j_{\mathsf{std}} = \arg\max_y s_y$ be the standard model's prediction. Because j_{std} is the $\arg\max$ of s, we have:

$$s_{j_{\mathsf{std}}} \geq s_{j_{\mathsf{ens}}}. \tag{1.43}$$

Taking the negation of this, we get:

$$-s_{j_{\mathsf{ens}}} \geq -s_{j_{\mathsf{std}}}. \tag{1.44}$$

Because j_{ens} is the $\arg\max$ of $r + s$, we have:

$$r_{j_{\mathsf{ens}}} + s_{j_{\mathsf{ens}}} \geq r_{j_{\mathsf{std}}} + s_{j_{\mathsf{std}}}. \tag{1.45}$$

Adding α times Inequality 1.45 with $(\alpha + \beta)$ times Inequality 1.44, we get:

$$\alpha r_{j_{\mathsf{ens}}} - \beta s_{j_{\mathsf{ens}}} \geq \alpha r_{j_{\mathsf{std}}} - \beta s_{j_{\mathsf{std}}}. \tag{1.46}$$

The rest of this step is the same as in the comparison between the ensemble and the robust model. Since softmax is monotonic, we have:

$$\text{softmax}(\alpha r - \beta s)_{j_{\mathsf{ens}}} \geq \text{softmax}(\alpha r - \beta s)_{j_{\mathsf{std}}}. \tag{1.47}$$

But from Equation 1.35 the LHS is the same as the robust model's probability of getting the label correct, and the RHS is the same as the ensemble's probability of getting the label correct:

$$P_{\mathsf{adv}}(Y = j_{\mathsf{ens}} \mid Z = z) \geq P_{\mathsf{adv}}(Y = j_{\mathsf{std}} \mid Z = z). \tag{1.48}$$

Taking negations (to get the error), and then the expectation over $Z = z$, we get (note that below we write the error, which is why the sign is now flipped):

$$\text{Err}_{\mathsf{ood}}(f_{\mathsf{std}}) \geq \text{Err}_{\mathsf{ood}}(f_{\mathsf{ens}}). \tag{1.49}$$

Which is what we wanted to show.

□

Dealing with class imbalance. Lemma 1.4, Equation 1.14 shows how to combine models in general, if the class-balanced assumption does not hold. Note the additional "$-m$" term. Here, the (marginal) probability of each class is defined in Equation 1.14.

(ID Analysis) Then, the "Proof of Proposition 4.1" is identical for the general case, we just need to set $f_{\mathsf{ens}}(x) = f_{\mathsf{std}}(x) + f_{\mathsf{rob}}(x) - m$ on the first line. Equation 1.28 then follows from Lemma 1.4, and the rest of the proof is identical.

(OOD Analysis) The OOD results, Proposition 4.2 and 4.3, follow if the class marginal distributions match up between ID and OOD, so $P_{\mathsf{id}}(Y = y) = P_{\mathsf{ood}}(Y = y)$. If the distribution over classes changes substantially, then ensembles can possibly do worse than the robust model.

2 MORE INFORMATION ON EXPERIMENTS

2.1 ADDITIONAL DETAILS ON DATASETS

Here we describe the robustness interventions and datasets in more detail.

Robustness interventions:

1. In-N-Out [Xie et al., 2021]. Many datasets contain a core input x (image or time series data), and metadata z (e.g., location or climate data). Xie et al. [2021] show that using the metadata (in addition to x) improves accuracy in-distribution (ID), but hurts accuracy out-of-distribution. Xie et al. [2021] consider a standard model that takes in both the core inputs and metadata to predict the target, and a robust model that only takes in the core inputs and does some additional pretraining. We use official checkpoints from their CodaLab worksheet `https://worksheets.codalab.org/worksheets/0x2613c72d4f3f4fbb94e0a32c17ce5fb0`, and compare to the results tagged as "In-N-Out" on each dataset. They also show results after doing additional self-training on (unlabeled) OOD data, but we do not compare to this because 1. OOD data is assumed to be unavailable in our setting, and 2. if OOD unlabeled data is available, we can also start from ID-calibrated ensembles and do additional self-training.
2. Lightweight fine-tuning [Kumar et al., 2022]: When adapting a pretrained model to an ID dataset, typically all the model parameters are fine-tuned. Recent works show that tuning only parts of the model can often do better OOD even though the ID performance is worse [Li and Liang, 2021, Houlsby et al., 2019]. On four distribution shift datasets, we take checkpoints from Kumar et al. [2022] where the standard model starts from a pretrained initialization and fine-tunes all parameters on an ID dataset, and the robust model only learns the top linear 'head' layer.
3. Zero-shot language prompting: Radford et al. [2021] pretrain a model on a large multi-modal language and vision dataset. The model can then predict the label of an image by comparing the image embedding, with the language embedding for prompts such as 'photo of an apple' or 'photo of a banana'. They show that this zero-shot language prompting approach (robust model) can be much more accurate OOD than the traditional method of fine-tuning the entire model (standard model), although ID accuracy of the robust model is worse. We use model checkpoints and datasets from Radford et al. [2021].
4. Group distributionally robust optimization (DRO) [Sagawa et al., 2020]: Standard ERM models often latch on to spurious correlations in a dataset, such as image background color, or the occurrence of certain words in a sentence. Group DRO essentially upweights examples where this spurious correlation is not present. The original formulation in Sagawa et al. [2020] assumes the spurious correlations are annotated, but newer variants [Liu et al., 2021] can work even without these annotations.
5. CORAL [Sun and Saenko, 2016] aims to align feature representations across different domains, by penalizing differences in the means and covariances of the feature distributions. The hope is that this generalizes better to OOD domains.

We consider three types of natural shifts (geography shifts, subpopulation shifts, style shifts), and we also consider adversarial spurious shifts.

Geography shifts. In geography shifts the ID data comes from some locations, and the OOD data comes from a different set of locations. One motivation is that in many developing areas training data may be unavailable because of monetary constraints [Jean et al., 2016].

1. **LandCover** [Rußwurm et al., 2020]: The goal is to classify a satellite images into one of 6 land types (e.g., "grassland", "savannas"). The ID data contains images from outside Africa, and the OOD data consists of images from Africa. We take model checkpoints from Xie et al. [2021] where they use the In-N-Out intervention—the core feature x is time series data measured by Nasa's MODIS satellite, and the spurious metadata z consists of climate data (e.g., temperature) at that location. We use the ID and OOD dataset splits defined by Xie et al. [2021].
2. **Cropland** [Wang et al., 2020]: The goal is to predict whether a satellite image is of a cropland or not. The ID dataset contains images from Iowa, Missouri, and Illinois, and the OOD dataset contains images from Indiana and Kentucky. We take model checkpoints from Xie et al. [2021] where they use the In-N-Out intervention—the core feature x is an RGB satellite image, and the spurious metadata z consists of location coordinates and vegetation bands. We use the ID and OOD dataset splits defined by Xie et al. [2021].
3. **iWildCam** [Beery et al., 2020, Koh et al., 2021]: The goal is to classify the species of an animal given a photo taken by a camera placed in the wild (e.g., in a forest). The ID dataset consists of photos taken by over 200 cameras,

and the OOD dataset consists of photos taken by held-out cameras. We use the splits by Koh et al. [2021]. We take model checkpoints from Koh et al. [2021], where the standard model is trained via standard empirical risk minimization (ERM), and the robust model is trained via CORAL. The model checkpoints were taken from `https://worksheets.codalab.org/worksheets/0x036017edb3c74b0692831fadfe8cbf1b`.

Subpopulation shifts. In subpopulation shifts, the ID data contains a few sub-categories (e.g., black bear and sloth bear), and the OOD data contains different sub-categories (e.g., brown bears and polar bears) or the same parent category (e.g., bears). For both datasets below, we take model checkpoints from Kumar et al. [2022] where they use the lightweight fine-tuning intervention, starting from a MoCo-v2 ResNet-50 model pretrained on unlabeled ImageNet images. The datasets are from Santurkar et al. [2020].

1. **Living-17** [Santurkar et al., 2020]: the goal is to classify an image as one of 17 animal categories such as “bear” - the ID dataset contains images of black bears and sloth bears and the OOD dataset has images of brown bears and polar bears.
2. **Entity-30** [Santurkar et al., 2020]: similar to Living-17, except the goal is to classify an image as one of 30 entity categories such as “food”, “motor vehicle”, and “index”.

Style shifts. In style shifts, the ID data contains data in a certain style (e.g., sketches), and the OOD data contains data in a different style (e.g., real photos, renditions).

1. **DomainNet** [Peng et al., 2019]: a standard domain adaptation dataset. Here, our ID dataset contains “sketch” images (e.g., drawings of apples, elephants, etc), and the OOD dataset contains “real” photos of the same categories. We take model checkpoints from Kumar et al. [2022] where they use the lightweight fine-tuning intervention, starting from a CLIP ResNet-50 model.
2. **CelebA** [Liu et al., 2015]: the goal is to classify a portrait of a face as “male” or “female” - the ID dataset contains images of people without hats, and the OOD dataset contains images of people wearing hats (some facial features might be “suppressed” or “missing” with hats). We take model checkpoints from Xie et al. [2021] where they use the In-N-Out intervention—the core feature x is the RGB image, and the spurious metadata z consists of 7 attribute tags annotated in the dataset (e.g., presence of makeup, beard).
3. **CIFAR->STL**: standard domain adaptation dataset [French et al., 2018], where the ID is CIFAR-10 [Krizhevsky, 2009], and the OOD is STL [Coates et al., 2011]. The task is to classify an image into one of 10 categories such as “dog”, “cat”, or “airplane”. We take model checkpoints from Kumar et al. [2022] where they use the lightweight fine-tuning intervention, starting from a MoCo-v2 ResNet-50 model pretrained on unlabeled ImageNet images.
4. **ImageNet** [Russakovsky et al., 2015]: a large scale dataset where the goal is to classify an image into one of 1000 categories. We use the zero-shot language prompting intervention using a CLIP ViT-B/16 vision transformer model. We evaluate on 3 standard OOD datasets: **ImageNetV2** [Recht et al., 2019],**ImageNet-R** [Hendrycks et al., 2020], and **ImageNet-Sketch** [Wang et al., 2019].

Adversarial spurious shifts. In adversarial spurious shifts, the ID dataset contains a feature that is correlated with a label, but this correlation is flipped OOD.

1. **Waterbirds** [Sagawa et al., 2020]: The goal is to classify an image as a “waterbird” or “landbird”. The dataset is synthetically constructed to have adversarially spurious features: “water” backgrounds are correlated with “waterbird” labels in the ID, but anticorrelated OOD. We use checkpoints from Jones et al. [2021] where they use the group DRO intervention.
2. **MNLI** [Williams et al., 2018]: The goal is to predict whether a hypothesis is entailed, contradicted by, or neutral to an associated premise. We use the splits in Sagawa et al. [2020]—they partition the dataset so that in-distribution “negation” words “nobody”, “no”, “never”, and “nothing” are correlated with the contradiction label, however in the OOD dataset these words are anticorrelated with the contradiction label. We use checkpoints from Jones et al. [2021] where they use the group DRO intervention.
3. **CivilComments** [Borkan et al., 2019]: The goal is to predict whether a comment is toxic or not. We use the splits in Sagawa et al. [2020]—they partition the dataset where in the ID split mentions of a Christian identity are correlated with non-toxic comments, but in the OOD split mentions of a Christian identity are correlated with a toxic comment. We use checkpoints from Jones et al. [2021] where they use the group DRO intervention. CivilComments is also used in Koh et al. [2021].

Table 1: *ID* accuracies: The in-distribution accuracies of calibrated ensembles, tuned ensembles, and vanilla ensembles are very close (within confidence intervals), so any of these methods are acceptable if we are looking at in-distribution accuracy. However, they perform quite differently when it comes to OOD accuracy (Table 2).

	Ent30	DomNet	CIFAR10	Liv17	Land	Crop	CelebA
Logits	93.7 (0.1)	89.3 (0.6)	**97.3 (0.1)**	**97.1 (0.2)**	**77.4 (0.1)**	95.5 (0.1)	93.4 (0.6)
Probs	**93.7 (0.1)**	89.1 (0.4)	**97.3 (0.1)**	**97.1 (0.2)**	**77.4 (0.2)**	95.5 (0.1)	93.4 (0.6)
Tuned Logits	**93.8 (0.0)**	**91.3 (0.2)**	**97.4 (0.1)**	97.1 (0.1)	**77.3 (0.4)**	**95.6 (0.1)**	**94.8 (0.2)**
Tuned Probs	93.8 (0.1)	90.6 (0.7)	**97.4 (0.1)**	**97.2 (0.1)**	77.1 (0.3)	95.5 (0.1)	**95.0 (0.2)**
Calibrated Logits	93.7 (0.1)	**91.1 (0.4)**	97.2 (0.1)	**97.2 (0.2)**	77.2 (0.2)	**95.6 (0.1)**	**94.5 (0.5)**
Calibrated Probs	93.7 (0.1)	**91.2 (0.7)**	97.2 (0.1)	**97.2 (0.2)**	77.2 (0.2)	**95.6 (0.1)**	**94.5 (0.5)**

	ImageNet	iWildCam	MNLI	Waterbirds	Comments
Logits	82.1 (-)	**84.2 (-)**	**82.9 (-)**	90.1 (-)	90.4 (-)
Probs	82.1 (-)	83.9 (-)	**82.9 (-)**	90.1 (-)	90.4 (-)
Tuned Logits	**82.7 (-)**	**84.1 (-)**	**83.0 (-)**	**93.2 (-)**	**92.7 (-)**
Tuned Probs	82.3 (-)	83.9 (-)	**83.0 (-)**	**93.2 (-)**	**92.6 (-)**
Calibrated Logits	82.0 (-)	**84.3 (-)**	82.8 (-)	92.9 (-)	91.4 (-)
Calibrated Probs	82.0 (-)	84.0 (-)	**82.8 (-)**	92.9 (-)	91.4 (-)

2.2 PER-DATASET RESULTS ON ENSEMBLING ABLATIONS

In Section 6 we ablated calibrated ensembles with “tuned” ensembles where the ensemble weights are tuned on in-distribution validation data, and with vanilla ensembles. Here, we show per-dataset results both ID (Table 1) and OOD (Table 2).

In Section 6, We also compared calibrated ensembles (of one standard and one robust model) with ensembles of two standard models, and ensembles of two robust models, where for a fair comparison all models are calibrated. We ran this ablation on 6 of the 14 datasets (Entity-30, DomainNet, CIFAR→STL, Living-17, Landcover, Cropland, and CelebA) because it requires multiple standard and multiple robust models, which were not available or very expensive to run on large datasets like ImageNet. Calibrated ensembles get an average ID accuracy of 91.8% (vs. 89.7% for a robust-robust ensemble and 90.7% for a standard-standard ensemble), and an average OOD accuracy of 76.5% (vs. 76.2% for a robust-robust ensemble and 68.8% for a standard-standard ensemble). We show per-dataset results in Table 3 (ID) and Table 4 (OOD). We show per-dataset results both ID (Table 3) and OOD (Table 4).

2.3 PER-DATASET RESULTS ON CALIBRATION AND CONFIDENCE

Relative confidence can be incorrect. We measure the confidence of a model f on a distribution P as $\text{conf}(f, P) = \mathbb{E}_{x \sim P}[\max_i f(x)_i]$. Even if the models are not calibrated OOD, one intuitive intuition for why calibrated ensembles work is that that robust model has higher confidence OOD, so that the ensemble primarily uses the (more accurate) robust model’s predictions OOD. However, on the remote sensing dataset Landcover we find that the robust model is 6% *less confident* on OOD data than the standard model even though the robust model is 5% *more accurate* OOD than the standard model. Interestingly, calibrated ensembles are able to combine the models in a more fine-grained way to get the best of both worlds, which is captured in our stylized setting in Section 4. We show the average confidence of the standard and robust models for each dataset ID (Table 7) and OOD (Table 8).

Per-dataset results for ECE. In Section 6.2, we talked about the ECE of the standard and robust models *after calibrating on ID data*. Here we show the results for each dataset ID (Table 5) and OOD (Table 6). We also show the ECE of the standard and robust models *before calibrating on ID data*, on ID (Table 9) and on OOD (Table 10).

References

Sara Beery, Elijah Cole, and Arvi Gjoka. The iwildcam 2020 competition dataset. *arXiv preprint arXiv:2004.10340*, 2020.

Table 2: *OOD* accuracies: calibrated ensembles outperform vanilla ensembles and even tuned ensembles where the combination weights are tuned to maximize in-distribution accuracy. Averaged across the datasets, calibrated ensembles get an OOD accuracy of 74.7%, while tuned ensembles get an accuracy of 72.1%. The in-distribution accuracies of the methods are very close (within 0.2% of each other).

	Ent30	DomNet	CIFAR10	Liv17	Land	Crop	CelebA
Logits	**64.9 (0.3)**	75.7 (1.2)	87.3 (0.2)	**81.8 (0.4)**	**60.5 (0.8)**	**90.9 (0.2)**	**76.9 (0.9)**
Probs	64.6 (0.4)	78.7 (1.3)	87.2 (0.2)	**81.8 (0.4)**	59.5 (1.0)	**90.9 (0.2)**	**76.9 (0.9)**
Tuned Logits	**64.6 (0.6)**	86.3 (0.6)	85.7 (0.9)	80.8 (0.7)	58.7 (1.2)	**87.3 (5.7)**	**77.5 (1.3)**
Tuned Probs	62.8 (0.7)	**86.9 (0.2)**	85.0 (1.3)	81.6 (0.5)	58.7 (2.2)	**86.8 (5.5)**	**77.6 (1.7)**
Calibrated Logits	**65.0 (0.4)**	84.4 (0.3)	**87.5 (0.2)**	**82.0 (0.4)**	**61.2 (0.8)**	**91.3 (0.8)**	**77.6 (1.2)**
Calibrated Probs	**64.7 (0.5)**	86.1 (0.2)	87.3 (0.2)	**82.2 (0.6)**	**60.8 (0.8)**	**91.3 (0.8)**	**77.6 (1.2)**

	ImNet-R	ImNet-V2	ImNet-Sk	iWildCam	MNLI	Waterbirds	Comments
Logits	73.1 (-)	**73.7 (-)**	52.1 (-)	**66.2 (-)**	73.1 (-)	66.9 (-)	**76.0 (-)**
Probs	77.5 (-)	73.4 (-)	52.0 (-)	65.3 (-)	72.4 (-)	66.9 (-)	**76.0 (-)**
Tuned Logits	64.7 (-)	**73.6 (-)**	47.9 (-)	66.0 (-)	68.0 (-)	**88.1 (-)**	60.3 (-)
Tuned Probs	64.0 (-)	72.6 (-)	45.5 (-)	65.3 (-)	69.4 (-)	**88.1 (-)**	61.5 (-)
Calibrated Logits	73.7 (-)	**73.6 (-)**	**52.3 (-)**	**66.1 (-)**	**73.6 (-)**	81.1 (-)	71.8 (-)
Calibrated Probs	**77.9 (-)**	73.2 (-)	**52.3 (-)**	**66.3 (-)**	73.2 (-)	81.1 (-)	71.8 (-)

Table 3: *ID* accuracies: Calibrated ensembles (one standard and one robust model) achieve comparable or better performance to Standard ensembles (ensemble of two calibrated standard models) and Robust ensembles (ensemble of two calibrated robust models).

	Ent30	DomNet	CIFAR10	Liv17	Land	CelebA
Std Ensemble	**94.0 (0.0)**	86.3 (0.4)	**97.7 (0.1)**	**97.0 (0.3)**	**77.9 (0.1)**	91.7 (0.4)
Rob Ensemble	90.9 (0.2)	89.3 (0.3)	92.0 (0.0)	**97.1 (0.1)**	73.4 (0.2)	**95.2 (0.4)**
Cal ensemble	93.7 (0.1)	**91.2 (0.7)**	97.2 (0.1)	**97.2 (0.2)**	77.2 (0.2)	94.5 (0.5)

Table 4: *OOD* accuracies: Calibrated ensembles (one standard and one robust model) achieve comparable or better performance to Standard ensembles (ensemble of two calibrated standard models) and Robust ensembles (ensemble of two calibrated robust models).

	Ent30	DomNet	CIFAR10	Liv17	Land	CelebA
Std Ensemble	61.7 (0.2)	57.9 (0.2)	83.5 (0.2)	78.6 (0.4)	57.5 (0.7)	73.7 (1.1)
Rob Ensemble	63.8 (0.4)	**87.5 (0.1)**	85.1 (0.1)	**82.4 (0.1)**	**60.5 (1.4)**	**78.0 (0.6)**
Cal ensemble	**64.7 (0.5)**	86.1 (0.2)	**87.3 (0.2)**	**82.2 (0.6)**	**60.8 (0.8)**	**77.6 (1.2)**

Table 5: *ID* ECE: The expected calibration error (ECE) of the standard and robust models on ID test data, after post-calibration in ID validation data. The ID calibration errors are low—note that we only use 500 examples to temperature scale, so for ImageNet we have fewer examples than classes for post-calibration, but the models are still fairly well calibrated.

	Ent30	DomNet	CIFAR10	Liv17	Land	Crop	CelebA	ImageNet	iWildCam	MNLI	Waterbirds	Comments
Cal. Standard	0.7 (0.1)	2.0 (0.3)	0.8 (0.2)	1.3 (0.2)	1.1 (0.5)	1.4 (0.3)	2.7 (0.4)	1.2 (-)	3.6 (-)	2.2 (-)	1.2 (-)	1.2 (-)
Cal. Robust	1.1 (0.4)	2.2 (0.2)	1.3 (0.2)	1.8 (0.0)	1.7 (0.3)	3.5 (0.2)	1.2 (0.3)	2.3 (-)	1.3 (-)	2.5 (-)	0.5 (-)	8.1 (-)

Table 6: *OOD* ECE: The expected calibration error (ECE) of the standard and robust models on OOD test data, after calibrating on ID validation data. The calibration errors here are high, especially compared to the ID calibration errors in Table 5.

	Ent30	DomNet	CIFAR10	Liv17	Land	Crop	CelebA	ImNet-R	ImNet-V2	ImNet-Sk	iWildCam	MNLI	Waterbirds	Comments
Cal. Standard	15.4 (0.8)	13.6 (1.5)	5.6 (1.1)	11.4 (0.3)	16.4 (0.8)	7.4 (4.8)	11.5 (1.0)	5.4 (-)	4.0 (-)	10.1 (-)	3.2 (-)	13.2 (-)	17.7 (-)	23.3 (-)
Cal. Robust	14.3 (1.5)	5.5 (0.5)	8.2 (0.0)	8.7 (0.2)	6.5 (1.1)	5.0 (0.3)	14.0 (1.4)	4.0 (-)	4.9 (-)	5.1 (-)	2.4 (-)	4.2 (-)	5.5 (-)	6.3 (-)

Table 7: *ID* Confidences: The confidence of the standard and robust models on ID test data (after calibrating on ID data). The standard model is typically more confidence than the robust model, which is reasonable since the standard model is also typically more accurate. There are a few exceptions such as DomainNet, CelebA, and WaterBirds where the standard model is less confident than the robust model, but the standard model is also less accurate in these cases, so this is also reasonable.

	Ent30	DomNet	CIFAR10	Liv17	Land	Crop	CelebA	ImageNet	iWildCam	MNLI	Waterbirds	Comments
Cal. Standard	93.1 (0.3)	83.7 (0.4)	96.9 (0.6)	97.0 (0.2)	76.5 (0.9)	95.5 (0.4)	91.7 (0.6)	82.1 (-)	82.1 (-)	82.6 (-)	87.9 (-)	93.6 (-)
Cal. Robust	89.9 (0.4)	89.6 (0.1)	91.0 (0.1)	96.0 (0.1)	71.3 (0.5)	94.9 (0.5)	94.7 (0.2)	68.1 (-)	82.3 (-)	81.9 (-)	93.2 (-)	87.0 (-)

Table 8: *OOD* Confidences. The confidence of the standard and robust models on OOD test data (after calibrating on ID data). The robust model is usually more confident than the standard model, which is reasonable since the robust model is also typically more accurate. However, Landcover is a noticable exception: the robust model is less confident OOD, even though it is more accurate (see Table 3).

	Ent30	DomNet	CIFAR10	Liv17	Land	Crop	CelebA	ImNet-R	ImNet-V2	ImNet-Sk	iWildCam	MNLI	Waterbirds	Comments
Cal. Standard	76.1 (0.8)	68.9 (1.5)	87.8 (1.2)	89.2 (0.5)	72.0 (1.9)	92.8 (1.0)	85.5 (1.5)	57.8 (-)	75.5 (-)	50.6 (-)	59.1 (-)	77.0 (-)	78.1 (-)	80.1 (-)
Cal. Robust	77.5 (0.4)	92.6 (0.4)	93.3 (0.1)	90.8 (0.2)	66.0 (0.6)	94.1 (0.4)	90.1 (0.1)	74.0 (-)	64.2 (-)	53.2 (-)	65.1 (-)	79.7 (-)	92.5 (-)	80.4 (-)

Table 9: *ID* ECE. The expected calibration error (ECE) of the standard and robust models on ID test data, *before calibration* (the key difference from Table 5 is that this is before calibration). We can see that calibration on ID substantially reduces the ECE on ID data (see Table 5)

	Ent30	DomNet	CIFAR10	Liv17	Land	Crop	CelebA	ImageNet	iWildCam	MNLI	Waterbirds	Comments
Standard	1.0 (0.1)	8.5 (0.7)	1.2 (0.1)	1.2 (0.1)	6.7 (1.2)	1.5 (0.3)	5.9 (0.5)	2.2 (-)	10.9 (-)	9.0 (-)	8.2 (-)	3.7 (-)
Robust	1.1 (0.3)	5.8 (1.3)	1.1 (0.2)	3.4 (0.4)	1.3 (0.1)	3.5 (0.1)	1.8 (0.2)	2.4 (-)	2.8 (-)	8.2 (-)	14.8 (-)	10.2 (-)

Table 10: *OOD* ECE: The expected calibration error (ECE) of the standard and robust models on OOD test data, *before calibration* (the key difference from Table 6 is that this is before calibration). The calibration errors here are higher than the ID calibration errors in Table 9. Comparing with Table 6 (which is after calibration on ID data), we see that calibrating ID does help OOD calibration a little, although the models still remain miscalibrated OOD.

	Ent30	DomNet	CIFAR10	Liv17	Land	Crop	CelebA	ImNet-R	ImNet-V2	ImNet-Sk	iWildCam	MNLI	Waterbirds	Comments
Standard	19.1 (0.3)	29.5 (0.5)	10.1 (0.3)	11.7 (0.4)	24.7 (1.5)	8.3 (4.3)	17.6 (0.5)	7.9 (-)	6.1 (-)	13.3 (-)	19.5 (-)	22.7 (-)	31.8 (-)	30.0 (-)
Robust	14.3 (1.6)	1.8 (0.8)	8.4 (0.3)	6.8 (0.2)	7.1 (1.3)	8.4 (0.7)	12.7 (0.7)	3.9 (-)	5.2 (-)	5.2 (-)	5.3 (-)	10.3 (-)	10.4 (-)	9.9 (-)

Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. Nuanced metrics for measuring unintended bias with real data for text classification. In *World Wide Web (WWW)*, pages 491–500, 2019.

Adam Coates, Andrew Ng, and Honlak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15, pages 215–223, 2011.

Geoff French, Michal Mackiewicz, and Mark Fisher. Self-ensembling for visual domain adaptation. In *International Conference on Learning Representations*, 2018.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. *arXiv preprint arXiv:2006.16241*, 2020.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. *arXiv*, 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.

Erik Jones, Shiori Sagawa, Pang Wei Koh, Ananya Kumar, and Percy Liang. Selective classification can magnify disparities across groups. In *International Conference on Learning Representations (ICLR)*, 2021.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton A. Earnshaw, Imran S. Haque, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. WILDS: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning (ICML)*, 2021.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *International Conference on Learning Representations (ICLR)*, 2022.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Association for Computational Linguistics (ACL)*, 2021.

Evan Zheran Liu, Behzad Haghgoo, Annie S. Chen, Aditi Raghunathan, Pang Wei Koh, Shiori Sagawa, Percy Liang, and Chelsea Finn. Just train twice: Improving group robustness without training group information. In *International Conference on Machine Learning (ICML)*, 2021.

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.

Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *International Conference on Computer Vision (ICCV)*, 2019.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, volume 139, pages 8748–8763, 2021.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning (ICML)*, 2019.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 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.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations (ICLR)*, 2020.

Shibani Santurkar, Dimitris Tsipras, and Aleksander Madry. Breeds: Benchmarks for subpopulation shift. *arXiv*, 2020.

Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European Conference on Computer Vision (ECCV)*, 2016.

Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

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.

Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Association for Computational Linguistics (ACL)*, pages 1112–1122, 2018.

Sang Michael Xie, Ananya Kumar, Robbie Jones, Fereshte Khani, Tengyu Ma, and Percy Liang. In-N-out: Pre-training and self-training using auxiliary information for out-of-distribution robustness. In *International Conference on Learning Representations (ICLR)*, 2021.