The Value of Out-of-Distribution Data

Ashwin De Silva 1 * Rahul Ramesh 2 * Carey E. Priebe 1 Pratik Chaudhari 2 † Joshua T. Vogelstein 1 †

Abstract

Generalization error always improves with more in-distribution data. However, it is an open question what happens as we add out-of-distribution (OOD) data. Intuitively, if the OOD data is quite different, it seems more data would harm generalization error, though if the OOD data are sufficiently similar, much empirical evidence suggests that OOD data can actually improve generalization error. We show a counter-intuitive phenomenon: the generalization error of a task can be a non-monotonic function of the amount of OOD data. Specifically, we show that generalization error can improve with small amounts of OOD data, and then get worse with larger amounts compared to no OOD data. In other words, there is value in training on small amounts of OOD data, and then get worse with larger amounts compared to no OOD data. In other words, we expect the target generalization error to be monotonic in the number of OOD samples; this is indeed the rationale behind classical works such as that of Ben-David et al. (2010) recommending against having OOD samples in the training data.

We show that a third counter-intuitive possibility occurs: OOD data from the same distribution can both improve or deteriorate the target generalization depending on the number of OOD samples. Generalization error (note: error, not the gap) on the target task is non-monotonic in the number of OOD samples. Across numerous examples, we find that there exists a threshold below which OOD samples improve generalization error on the target task but if the number of OOD samples is beyond this threshold, then the generalization error deteriorates. To our knowledge, this phenomenon has not been predicted or demonstrated by any other theoretical or empirical result in the literature.

We first demonstrate the non-monotonic behavior through a simple but theoretically tractable problem using Fisher’s Linear Discriminant (FLD). In §3.3, for the same problem, we compare the actual expected target generalization error with the theoretical upper bound developed by (Ben-David et al., 2010) to show that this phenomenon is not captured by existing theory. We also present empirical evidence for the presence of non-monotonic trends in target generalization error, on tasks and experimental settings constructed from the MNIST, CIFAR-10, PACS and DomainNet datasets. Our code is available at https://github.com/neurodata/value-of-ood-data.

1. Introduction

Real data is often heterogeneous and more often than not, suffers from distribution shifts. We can model this heterogeneity as samples drawn from a mixture of a target distribution and from “out-of-distribution” (OOD). For a model trained on such data, we expect one of the following outcomes: (i) if the OOD data is similar to the target data, then more OOD samples will help us generalize to the target distribution; (ii) if the OOD data is dissimilar to the target data, then more samples are detrimental. In other words, we expect the target generalization error to be monotonic in the number of OOD samples; this is indeed the rationale behind classical works such as that of Ben-David et al. (2010) recommending against having OOD samples in the training data.

We show that a third counter-intuitive possibility occurs: OOD data from the same distribution can both improve or deteriorate the target generalization depending on the number of OOD samples. Generalization error (note: error, not the gap) on the target task is non-monotonic in the number of OOD samples. Across numerous examples, we find that there exists a threshold below which OOD samples improve generalization error on the target task but if the number of OOD samples is beyond this threshold, then the generalization error deteriorates. To our knowledge, this phenomenon has not been predicted or demonstrated by any other theoretical or empirical result in the literature.

We first demonstrate the non-monotonic behavior through a simple but theoretically tractable problem using Fisher’s Linear Discriminant (FLD). In §3.3, for the same problem, we compare the actual expected target generalization error with the theoretical upper bound developed by (Ben-David et al., 2010) to show that this phenomenon is not captured by existing theory. We also present empirical evidence for the presence of non-monotonic trends in target generalization error, on tasks and experimental settings constructed from the MNIST, CIFAR-10, PACS and DomainNet datasets. Our code is available at https://github.com/neurodata/value-of-ood-data.

1.1. Outlook

Consider the idealistic setting where we know which samples in the dataset are OOD. A trivial solution could be to remove the OOD samples from the training set. But the fact that the generalization error is non-monotonic also suggests a better solution. We show on a number of benchmark
tasks that by using an appropriately weighted objective between the target and OOD samples, we can ensure that the generalization error on the target task decreases monotonically with the number of OOD samples. This is merely a proof-of-concept for this idealistic setting. But it does suggest that if one could detect the OOD samples, then there are not only ways to safeguard against them but there are also ways to benefit from them.

Of course, we do not know which samples are OOD in real datasets. When datasets are curated incrementally, the fraction of OOD samples can also change with time, and the implicit benefit of these OOD data may become a drawback later. When we do not know which samples are OOD, we show how a number of go-to strategies such as data-augmentation, hyper-parameter optimization and pre-training the network are not enough to ensure that the generalization error on the target does not deteriorate with the number of OOD samples.

Our results indicate that non-monotonic trends in generalization error are a significant concern, especially when the presence of OOD samples in the dataset goes undetected. The main contribution of this paper is to highlight the importance of this phenomenon. We leave the development of a practical solution for future work.

2. Generalization error is non-monotonic in the number of OOD samples

We define a distribution $P$ as a joint distribution over the input domain $X$ and the output domain $Y$. We model the heterogeneity in the dataset as two distributions: $n$ samples drawn from a target distribution $P_t$ and $m$ samples drawn from out-of-distribution (OOD) $P_o$. We would like to minimize the generalization error $e_t(h) = \mathbb{E}_{(x,y) \sim P_t}[h(x) \neq y]$ on the target distribution. Suppose we assume that all the data comes from a single target distribution because we are unaware of the presence of OOD samples in the dataset. Therefore, we may find a hypothesis that minimizes the empirical loss

$$\hat{e}(h) = \frac{1}{n + m} \sum_{i=1}^{n+m} \ell(h(x_i), y_i),$$

using the dataset $\{(x_i, y_i)\}_{i=1}^{n+m}$, here $\ell$ measures the mismatch between the prediction $h(x_i)$ and label $y_i$. If $P_t = P_o$, then $e_t(h) - \hat{e}(h) = \mathcal{O}\left((n + m)^{-1/2}\right)$ (Smola & Schölkopf, 1998). But if $P_t$ is far enough from $P_o$ in certain ways, then we expect that the error on $P_t$ of a hypothesis obtained by minimizing the average empirical loss will be suboptimal, especially when the number of OOD samples $m \gg n$.

2.1. An example using Fisher’s Linear Discriminant

Consider a binary classification problem with one-dimensional inputs in Figure 1. Target samples are drawn from a Gaussian mixture model (with means $\{-\mu + \Delta, \mu + \Delta\}$; see Appendix A.1 for details. Fisher’s linear discriminant (FLD) is a linear classifier for binary classification problems and it computes $h(x) = 1$ if $\omega^T x > c$ and $h(x) = 0$ otherwise; here $\omega$ is a projection vector which acts as a feature extractor and $c$ is a threshold that performs one-dimensional discrimination between the two classes. FLD is optimal when the class conditional density of each class is a multivariate Gaussian distribution with the same covariance structure. We provide a detailed account of FLD in Appendix A.2.

Suppose we fit an FLD on a dataset which comprises of $n$ target samples and $m$ OOD samples. Also, suppose we do not know which samples are OOD and believe that all the samples in the dataset come from a single target distribution. For univariate data with equal class priors, the FLD decision rule reduces to,

$$\hat{h}(x) = \begin{cases} 1, & x > \frac{\hat{\mu}_0 + \hat{\mu}_1}{2} \\ 0, & \text{otherwise} \end{cases}$$

Define the decision threshold to be $\hat{c} = (\hat{\mu}_0 + \hat{\mu}_1)/2$. We can calculate (Appendices A.2 and A.3) an analytical expression

\begin{center}
\begin{tabular}{c|c|c}
\hline
\textbf{OOD Translation ($\Delta$)} & \textbf{Generalization Error} \\
\hline
0 & 0.0000 \\
1 & 0.0004 \\
1.2 & 0.0008 \\
1.4 & 0.0009 \\
1.6 & 0.0009 \\
\hline
\textbf{OOD Translation ($\Delta$)} & \textbf{Generalization Error} \\
\hline
2.0 & 0.0020 \\
\hline
\end{tabular}
\end{center}

Figure 1. Left: A schematic of the Gaussian mixture model corresponding to the target (top) and OOD samples (bottom). The OOD sample size ($m = 28$) at which the target generalization error is minimized at $\Delta = 1.6$ is indicated at the top. Right: For $n = 100$, we plot the generalization error of FLD on the target distribution as a function of the ratio of OOD and target samples $m/n$, for different types of OOD samples corresponding to different values of $\Delta$. This plot uses the analytical expression for the generalization error in (2); see Appendix A.6 for a numerical simulation study. For small values of $\Delta$, when the two distributions are similar to each other, the generalization error $e_t(h)$ decreases monotonically. However, beyond a certain value of $\Delta$, the generalization error is non-monotonic in the number of OOD samples. The optimal value of $m/n$ which leads to the best generalization error is a function of the relatedness between the two distributions, as governed by $\Delta$ in this example. This non-monotonic behavior can be explained in terms of a bias-variance tradeoff with respect to the target distribution: a large number of OOD samples reduces the variance but also results in a bias with respect to the optimal hypothesis of the target.
for the generalization error of FLD on the target distribution:
\[
e_{b}(\hat{h}) = \frac{1}{2} \Phi \left( \frac{m \Delta - (n + m) \mu}{\sqrt{(n + m)(n + m + 1)}} \right) + \Phi \left( - \frac{m \Delta - (n + m) \mu}{\sqrt{(n + m)(n + m + 1)}} \right).
\]
here \( \Phi \) is the CDF of the standard normal distribution.

Figure 2. Mean squared error (MSE) (Y-axis) of the decision threshold \( \hat{c} \) of FLD (see Appendix A.3), for the same setup as that of Figure 1, plotted against the ratio of the OOD and target samples \( m/n \) (X-axis) for \( \Delta = 1.8 \). Squared bias and variance of the MSE are in violet and blue, respectively. This illustration clearly demonstrates the intuition behind non-monotonic target error: the MSE drops initially because of the smaller variance due to the OOD samples. With more OOD samples, MSE increases due to the increasing bias. Non-monotonic trend in MSE of \( \hat{c} \) translates to a similar trend in the target generalization error (0-1 loss).

Figure 3. We can control the Bayes optimal error by adjusting \( \mu, \sigma \) of the Gaussian mixture model in \$2.1$. As discussed in Remark 2, when the Bayes optimal error is large for \( \mu = 6, \sigma = 16 \), we can observe non-monotonic trends even for a large number of target samples \( n = 500 \). This suggests that non-monotonic trends in generalization are not limited to small sample sizes.

Figure 1 (right) shows how the generalization error \( e_{b}(\hat{h}) \) decreases up to some threshold of the ratio between the number of OOD samples and the number of target samples \( m/n \) and then increases beyond that. This threshold is different for different values of \( \Delta \) as one can see in (2) and Figure 1 (right). This behavior is surprising because one would \textit{a priori} expect the generalization error to be monotonic in the number of OOD samples. The fact that a non-monotonic trend is observed even for a one dimensional Gaussian mixture model suggests that this may be a general phenomenon. We can capture this discussion as a theorem; the FLD example above is the proof.

**Theorem 1.** There exist target and OOD distributions, \( P_t \) and \( P_o \) respectively, such that the generalization error on the target distribution of the hypothesis that minimizes the empirical loss in (1), is non-monotonic in the number of OOD samples. In particular, there exist distributions \( P_t \) and \( P_o \) such that the generalization error decreases with few OOD samples and increases with even more OOD samples, compared to no OOD samples.

**Remark 2 (An intuitive explanation of non-monotonic trends in generalization error).** Suppose that a learning algorithm achieves Bayes optimal error on the target distribution with high probability when the target sample size \( n \) exceeds \( N \). We argue that a non-monotonic trend in generalization error is likely to occur when \( n < N \), i.e., when target generalization error is higher than the Bayes optimal error. In this case, if we add OOD samples whose empirical distribution is sufficiently close to that of the target distribution, then this would improve generalization by reducing the variance of the learned hypothesis. But as the OOD sample size increases, the difference between the two distributions becomes apparent and this leads to a bias in the choice of the hypothesis. Figure 2 illustrates this phenomenon with regards to our FLD example in Figure 1, by plotting the mean squared error of the decision threshold \( \hat{c} \) and its constituent bias and variance components. Roughly speaking, we may understand the non-monotonic trend in generalization as a phenomenon that arises due to the finite number of OOD samples \( m/n \) in the example above). The distance between the distribution of the OOD samples and the distribution of the target samples \( \Delta \) in the example) determines the threshold beyond which the error is monotonic. Current tools in learning theory (Smola & Schölkopf, 1998) are fundamentally about understanding generalization when the number of samples is asymptotically large—whether they be from the target or OOD. In future work, we hope to formally characterize this non-monotonic trend in generalization error by building new learning-theoretic tools.

Even if the non-monotonic trend occurs for relatively small values of target and OOD samples \( n \) and \( m \) respectively in Figure 1, this need not always be the case. **If the number of samples \( N \) required to reach Bayes optimal error in the above remark is large, then a non-monotonic trend can occur even for large target sample size \( n \) (see Figure 3).**

**2.2. Non-monotonic trends for neural networks and machine learning benchmark datasets**

We experiment with several popular datasets including MNIST, CIFAR-10, PACS, and DomainNet and 3 different network architectures: (a) a small convolutional network with 0.12M parameters (denoted by SmallConv), (b) a wide residual network (Zagoruyko & Komodakis, 2016) of depth 10 and widening factor 2 (WRN-10-2), and (c) a larger wide residual network of depth 16 and widening factor 4 (WRN-16-4). See Appendix B.4 for more details.

**A non-monotonic trend in generalization error can occur due to geometric and semantic nuisances.** Such nu-
sances are very common even in curated datasets (Van Horn, 2019). We constructed 5 binary classification sub-tasks (denoted by $T_i$ for $i = 1, \ldots, 5$) from CIFAR-10 to study this aspect (see Appendix B.1). We consider a CIFAR-10 sub-task $T_2$ (Bird vs. Cat) as the target and introduce rotated images by a fixed angle between $0^\circ$-$135^\circ$ as OOD samples. Figure 4 (left) shows that the generalization error decreases monotonically for small rotations but it is non-monotonic for larger angles. Next, we considered the sub-task $T_3$ (Frog vs. Horse) as the target distribution and generate OOD samples by adding Gaussian blur of varying levels to images from the same distribution. In Figure 4 (middle), the generalization error on the target is a monotonically decreasing function of the number of OOD samples for low blur but it increases non-monotonically for high blur.

Non-monotonic trends can occur when OOD samples are drawn from a different distribution. Large datasets can contain categories whose appearance evolves in time (e.g., a typical laptop in 2022 looks very different from that of 1992), or categories can have semantic intra-class nuisances (e.g., chairs of different shapes). We use 5 CIFAR-10 sub-tasks to study how such differences can lead to non-monotonic trends (see Appendix B.1). Each sub-task is a binary classification problem with two consecutive classes: Automobile vs. Motorcycle, Bird vs. Cat, etc. We consider $(T_i, T_j)$ as the (target, OOD) pair and evaluated the trend in generalization error for all 20 distinct pairs of distributions. Figure 4 (right) illustrates non-monotonic trends for 3 such pairs; see Appendix B for more details.

Non-monotonic trends also occur for benchmark domain generalization datasets. We further investigated three widely used benchmarks in the domain generalization literature. First, we consider the Rotated MNIST benchmark from DomainBed (Gulrajani & Lopez-Paz, 2020). We define the 10-way classification of un-rotated MNIST images as the target distribution and $\theta$-rotated MNIST images as the OOD samples. Similar to the previous rotated CIFAR-10 experiment, we observe non-monotonic trends in target generalization for larger angles $\theta$. Next, we consider the PACS benchmark from DomainBed which contains 4 distinct environments: photo, art, cartoon, and sketch. A 3-way classification task involving photos (real images) is defined as the target distribution, and we let the corresponding data from other environments be the OOD samples. Interestingly, we observe that when OOD samples consist of sketched images, then the generalization error on the real images exhibits a non-monotonic trend. We also observe similar trends in DomainNet, a benchmark that resembles PACS; see Figure 5.

Generalization error is not always non-monotonic even when there is distribution shift. We considered CINIC-10 (Darlow et al., 2018), a dataset which was created by combining CIFAR-10 with images selected and down-sampled from ImageNet. We train a network on a subset of CINIC-10 that comprises of both CIFAR-10 and ImageNet images. The target task is CIFAR-10 itself, so images from ImageNet in CINIC-10 act as OOD samples. Figure 6 demonstrates that having more ImageNet samples in the training data im-

Figure 4. Left: Sub-task $T_3$ (Bird vs. Cat) from Split-CIFAR10 is the target data and images of these classes rotated by different angles $\theta$ are the OOD data. WRN-10-2 architecture was used to train the model. We see non-monotonic curves for larger values of $\theta$. For $60^\circ$ and $135^\circ$ in particular, the generalization error at $m/n = 20$ is worse than the generalization error with a fewer OOD samples, i.e. OOD samples actively hurt generalization. See Figure A8 (left) for a similar experiment with SmallConv.

Middle: The Split-CIFAR10 binary sub-task $T_4$ (Frog vs. Horse) is the target distribution and images with different levels of Gaussian blur are the OOD samples. WRN-10-2 architecture was used to train the model. Non-monotonic curves are observed for larger levels of blur, while for smaller levels of blur, we notice that adding more OOD data improves the generalization on the target distribution.

Right: Generalization error of two separate networks, WRN-10-2 and SmallConv, on the target distribution is plotted against the number of OOD samples for 3 different target-OOD pairs from Split-CIFAR10. All the 3 pairs exhibit non-monotonic target generalization trends across both network models. See Appendices B.2 and B.3 for experimental details and Appendix B.6 for experiments on more target-OOD pairs (Figures A6 and A7) and multiple target sample sizes (Figure A5). Error bars indicate 95% confidence intervals (10 runs).
To evaluate whether these three techniques work, we used the CIFAR-10 sub-task $T_2$ (Bird vs. Cat) as the target distribution and $T_5$ (Ship vs. Truck) as the distribution of the OOD data and trained a WRN-10-2 network under various settings. The results are reported in Figure 7; we find that these techniques do not mitigate the deterioration of target generalization error as the number of OOD samples in the dataset increases.

Effect of the target sample size on non-monotonicity
Unlike our previous experiments where we fixed the target sample size, in Figure 8 we plot the target error as we change both target and OOD sample sizes across 3 different fixed target-OOD pairs. The target generalization error is non-monotonic in the number of OOD samples when we have a small number of target samples for all target-OOD pairs (the solid dark lines that “dip” first before increasing later). However, as the number of target samples increases, the non-monotonicity is less pronounced or even completely absent. When we have a large number of target samples, the model is closer to the Bayes error and benefits less from more OOD. Although we do not observe this in Figure 8, we believe that Remark 2 that non-monotonicity could theoretically occur even at large target sample sizes, if the number of samples required to attain the Bayes optimal error is high.

3. Can we exploit the non-monotonic trend in the generalization error?
Assumption in Sections 3.1 and 3.2 In the previous section, we discussed non-monotonic trends in generalization error due to the presence of OOD samples in training datasets. If we do not know which samples are OOD, then...
The Value of Out-of-Distribution Data

Figure 7. **Left:** For the CIFAR-10 sub-task $T_2$ (Bird vs Cat) as target and $T_5$ (Ship vs Truck) as OOD, we train a WRN-10-2 network with class-balanced datasets with fixed number of target samples ($n = 100$) and different number ($m$) of OOD samples, under the following settings: (1) Vanilla, i.e., without any data-augmentation or pre-training (darkest red), (2) Data augmentation by padding, random cropping and random left/right flips (medium red), and (3) Pre-training followed by fine-tuning (lightest red). We pre-train the network on the 14000 class-balanced ImageNet images from CINIC-10 (see Appendix B.1) belonging to Bird and Cat classes which correspond to our hypothetical target distribution. Pre-training is performed for 100 epochs with a learning rate of 0.01. Next, we employ a two-step strategy of linear probing (first 50 epochs) and full-fine tuning (last 50 epochs) inspired by (Kumar et al., 2022) at a reduced learning rate of 0.001. Note that this fine-tuning is performed on the combined dataset of $n$ target and $m$ OOD samples. Even though data augmentation and pre-training followed by fine-tuning reduce the overall error, the generalization error still deteriorates as the fraction of OOD sample in the dataset increases. **Right:** For each value of $m$, we perform hyper-parameter tuning using Ray (Liang et al., 2018) over a validation set that has only target samples, and record the target generalization error of the model using the best set of hyper-parameters. We still observe deterioration of the target generalization error as the OOD samples increase. Note that such hyper-parameter tuning cannot be implemented in reality because we may not know the identity of the target and OOD samples. So the fact that the non-monotonic trend persists in the hypothetical instance where we know the sample identities guarantees that it will occur in practice as well. Error bars indicate 95% confidence intervals over 10 experiments.

The generalization error is bounded above by the following inequality

$$
\alpha \hat{e}_\alpha(h) \leq \epsilon(h) + \sqrt{\frac{\alpha^2}{n} + \frac{(1-\alpha)^2}{m}} \sqrt{V_H - \log\delta + 2(1-\alpha)d_H(P_t, P_o)},
$$

with probability at least $1-\delta$. Here $h^* = \text{argmin}_{h \in H} \epsilon_t(h)$ is the target error minimizer; $V_H$ is a constant proportional to the VC-dimension of the hypothesis class $H$ and $d_H(P_t, P_o)$ is a notion of relatedness between the distributions $P_t$ and $P_o$.

In other words, if we use an appropriate value of $\alpha$ that makes the second and third terms on the right-hand side small, then we can mitigate the deterioration of generalization error due to OOD samples. If the OOD samples are very different from the target samples, i.e., if $d_H(P_t, P_o)$ is large, then this theorem suggests that we should pick an $\alpha \approx 1$. Doing so effectively ignores the OOD samples and the generalization error then decreases monotonically as $O(n^{-1/2})$. Note that computation and minimization of the $\alpha$-weighted convex combination of target and OOD losses, $\alpha \hat{e}_\alpha(h) + (1-\alpha)\hat{e}_0(h)$, is possible only when the identities of target and OOD samples are known in advance.

### 3.1. Choosing the optimal $\alpha^*$

If we define $\rho = \frac{\sqrt{\alpha^2/n + (1-\alpha)^2/m}}{d_H(P_t, P_o)}$ to be, roughly speaking, the ratio of the capacity of the hypothesis class and the distance between distributions, then a short calculation shows that
for $\alpha \in [0, 1]$,

$$\alpha^* = \begin{cases} 
\frac{1}{n^{\frac{1}{2}}} \left( 1 + \sqrt{\frac{m^2}{4\rho^2(n+m)-mn}} \right) & \text{if } n \geq 4\rho^2, \\
\text{else.} & 
\end{cases}$$

This suggests that if we have a hypothesis space with small VC-dimension or if the OOD samples and target samples come from very different distributions, then we should train only on the target samples to obtain optimal error. Otherwise, including the OOD samples after appropriately weighing them using $\alpha^*$ can give a better generalization error.

It is not easy to estimate $\rho$ because it depends upon the VC-dimension of the hypothesis class (Ben-David et al., 2010; Vedantam et al., 2021). But in general, we can treat $\alpha$ as a hyperparameter and use validation data to search for its optimal value. For our FLD example we can do slightly better: we can calculate the analytical expression for the generalization error for the hypothesis that minimizes the $\alpha$-weighted empirical loss (see Appendices A.4 and A.5) and calculate $\alpha^*$ by numerically evaluating the expression for $\alpha \in [0, 1]$.

![Figure 9. Left: Generalization error on the target distribution for the Gaussian mixture model using a weighted objective (Theorem 3) in FLD; see Appendix A.4. Note that unlike in Figure 1, the generalization error monotonically decreases with the number of OOD samples $m$. Right: The optimal $\alpha^*$ that yields the smallest target generalization error as a function of the number of OOD samples. Note that $\alpha^*$ increases as the number of OOD samples $m$ increases; this increase is more drastic for large values of $\Delta$ and is more gradual for small values of $\Delta$. Observe that $\alpha^* = 1/2$ for all values of $m$ if $\Delta = 0$. See Appendix A.6 for a numerical simulation.]

Figure 9 shows that regardless of the number of OOD samples, $m$, and the relatedness between OOD and target, $\Delta$, we can obtain a generalization error that is always better than that of a hypothesis trained without OOD samples. In other words, if we choose $\alpha^*$ appropriately (Figure 1 corresponds to choosing $\alpha = 1/2$), then we do not suffer from non-monotonic generalization error on the target distribution.

### 3.2. Training networks with the $\alpha$-weighted objective

In §2.2, for a variety of computer vision datasets, we found that for some target-OOD pairs, the generalization error is non-monotonic in the number of OOD samples. We now show that if we knew which samples were OOD, then we can rectify this trend using an appropriate value of $\alpha^*$ to weigh the samples differently. In Figure 10, we track the test error of the target distribution for three cases: training is agnostic to the presence of OOD samples (red), the learner knows which samples are OOD and uses an $\alpha = 1/2$ in the weighted loss to train (yellow, we call this “naive”), and when it uses an optimal value of $\alpha$ using grid-search (green). Searching over $\alpha$ improves the test error on all these 3 ptarget-OOD pairs.

We also conducted another experiment to check if augmentation can help rectify the non-monotonic trend in the generalization error, using the $\alpha$-weighted objective, i.e., when we know which samples are OOD. As shown in Figure 11, in this case even naively weighing the objective ($\alpha = 1/2$, yellow) can rectify the non-monotonic trend, using the optimal $\alpha^*$ (green) further improves the error. This suggests that augmentation is an effective way to mitigate non-monotonic behavior, but only if we use the $\alpha$-weighted objective, which requires knowing which samples are OOD.

As we discussed in Figure 7, if we do not know which samples are OOD, then augmentation does not help.

#### Sampling mini-batches during training

For $m \gg n$, mini-batches that are sampled uniformly randomly from the dataset will be dominated by OOD samples. As a result, the gradient even if it is still unbiased, is computed using very few target samples. This leads to an increase in the test error, which is particularly noticeable with $\alpha^*$ chosen appropriately after grid search. We therefore use a biased sampling procedure where each mini-batch contains a fraction $\beta$ target samples and the remainder $1 - \beta$ consists of OOD samples. This parameter controls the bias and variance of the gradient of the target loss ($\beta = \frac{n}{n+m}$ gives unbiased gradients with respect to the unweighted total objective and high variance with respect to the target loss when $m \gg n$, see Appendix B.5). We found that both $\beta = \{0.5, 0.75\}$ improve test error.

#### Weighted objective for over-parameterized networks

It has been argued previously that weighted objectives are not effective for over-parameterized models such as deep networks because both surrogate losses $\hat{e}_t(h)$ and $\hat{e}_o(h)$ are zero when the model fits the training dataset (Byrd & Lipton, 2019). It may therefore seem that the weighted objective in Theorem 3 cannot help us mitigate the non-monotonic nature of the generalization error; indeed the minimizer of $\alpha \hat{e}_t(h) + (1-\alpha)\hat{e}_o(h)$ is the same for any $\alpha$ if the minimum is exactly zero. Our experiments suggest otherwise: the value of $\alpha$ does impact the generalization error—even for deep networks. This is perhaps because even if the cross-entropy loss is near-zero for a deep network towards the end of training, it is never exactly zero.
Figure 10. Here we present three settings: minimizing the average loss over target and OOD samples is agnostic to OOD samples present (red), minimizing the sum of the average loss of the target and OOD samples which corresponds to $\alpha = 1/2$ (yellow), minimizing an optimally weighted convex combination of the target and OOD empirical loss (green). The last two settings are only possible when one knows which samples are OOD. For each setting, we plot the generalization error on the target distribution against the number of OOD samples for (target, OOD) pairs from PACS (Left) and CIFAR-10 sub-tasks (Middle). Unlike in CIFAR-10 task pairs, we observe that in PACS, the target generalization error has a downward trend when $\alpha = 0.5$ (yellow line, left panel). We speculate that this could be due to the similarity between the target and OOD samples, which causes the model to generalize to the target even at a naive setting. Right: The optimal $\alpha^*$ obtained via grid search for the three problems in the middle column plotted against different number of OOD samples. The value of $\alpha^*$ lies very close to 1 but it is never exactly 1. In other words, if we use the weighted objective in Theorem 3 then we always obtain some benefit, even if it is marginal when OOD samples are very different from those of the target. Error bars indicate 95% confidence intervals over 10 experiments.

Limitations of the proof-of-concept solution The numerical and experimental evidence above indicate that even a weighted empirical risk minimization (ERM) algorithm between the target and OOD samples is able to rectify the non-monotonicity. However, this procedure is dependent on two critical ideal conditions: (1) We must know which samples in the dataset are OOD, and (2) We must have a held out dataset of target samples to tune the weight $\alpha$. The difficulty of meeting both of these conditions in reality limits the utility of this procedure as a practical solution to the problem. Instead, we hope that it would serve as a proof-of-concept solution that motivates future research into accurately identifying OOD samples within datasets, designing ways of determining the optimal weights, and developing better procedures for exploiting OOD samples to achieve a lower generalization.

3.3. Does the upper bound in Theorem 3 inform the non-monotonic trends?

Theorem 3 formed the basis for a proof-of-concept solution in an idealistic setting that exploits OOD samples to reduce target generalization error and effectively correct the non-monotonic trend. Next, we study whether this upper bound predicts the non-monotonic trend.

We return to the setting where we are unaware of the presence of OOD samples in the dataset, and minimize (1), assuming that all data comes from a single target distribution. We then apply Theorem 3 to our FLD example to derive the following upper bound $U = U(n, m, \Delta)$ for expected error on the target distribution:

$$U = \Phi(-\mu) + \Phi(\frac{\Delta - \mu}{\sigma}) + \frac{2m\left(\frac{d_H(\Delta)}{2} + \lambda\right)}{n + m}$$

where $\lambda = \Phi(-\frac{\Delta - \mu}{\sigma}) + \Phi(\frac{\Delta - \mu}{\sigma})$. The derivation (including the procedure of numerically computing $d_H(\Delta)$) is given in the Appendix A.7. Figure 12 compares the value of the upper bound $U$ with the actual expected target error $e_t(\hat{h})$ computed using (2).

Figure 11. Effect of data augmentation (padding with random cropping and random left/right flipping). Although the network trained in the setting where the OOD sample identities are unknown (red) continues to perform poorly with lots of OOD samples, even a naive weighing of the target and OOD loss ($\alpha = 1/2$) is enough to provide a monotonically decreasing error (yellow) when the OOD sample identities are known. This suggests that data augmentation may mitigate some of the anomalies that arise from OOD data, although we can do better by addressing them specifically using, for instance, the weighted objective (green). Error bars indicate 95% confidence intervals over 10 experiments.

Figure 12. Here we plot the true expected target error (bottom) and the generalization error upper bound value (top) against the $m/n$ ratio for the FLD example ($\mu = 5, \sigma = 10$) in Figure 1. The upper bound is significantly vacuous and does not follow the non-monotonic trend of the true target error. However, there are situations when the shape of the upper bound curve is consistent with that of true error (e.g., for large values of shift $\Delta$ between distributions of the target and OOD data). These observations are reported in Appendix A.8.

The upper bound in Figure 12 is vacuous and does not fol-
low a non-monotonic trend when the true error does. Even
though its shape fairly agrees with that of true error when $n$
and $\Delta$ are high, it fails to capture the non-monotonic trend
we have identified in §2.1. The fact that it eludes the grasp of
existing theory points to the counter-intuitive nature of this
observation and a need for a theoretical investigation of this
phenomenon. See Appendix A.8 for more comparisons.

4. Related Work and Discussion

Distribution shift (Quinonero-Candela et al., 2008)
and its variants such as covariate shift (Ben-David &
Urner, 2012; Reddi et al., 2015), concept drift (Mohri &
Muñoz Medina, 2012; Bartlett, 1992; Cavallanti et al., 2007),
domain shift (Gulrajani & Lopez-Paz, 2020; Sagawa et al.,
2021; Ben-David et al., 2010), sub-population shift (San-
turkar et al., 2020; Hu et al., 2018; Sagawa et al., 2019),
data poisoning (Yang et al., 2017; Steinhardt et al., 2017),
geometric and semantic nuisances (Van Horn, 2019), and
flawed annotations (Fréney & Verleysen, 2013) can lead
to the presence of OOD samples in a curated dataset, and
thereby may yield sub-optimal generalization error on the
desired task. While these problems have been studied in
the sense of an out-of-domain distribution, we believe that
we have identified a fundamentally different phenomenon,
namely a non-monotonic trend in the generalization error
with respect to the OOD samples in training data.

Internal Dataset Shift A recent body of works (Kaplun
et al., 2022; Swayamdipta et al., 2020; Siddiqui et al., 2022;
Jain et al., 2022; Maini et al., 2022) has investigated the
presence of noisy, hard-to-learn, and/or negatively influen-
tial samples in popular vision benchmarks. Existence of
such OOD samples indicates that the internal dataset shift
may be a widespread problem in real datasets. Such circum-
stances may give rise to undesired non-monotonic trends in
generalization error, as we have described in our work.

Domain Adaptation While most works listed above pro-
vide attractive ways of adapting or being robust to various
modes of shift, a part of our work addresses the question: if
we know which samples are OOD, can we optimally uti-
lize them to achieve a better generalization on the desired
target task? This is related to domain adaptation (Ben-David
et al., 2010; Mansour et al., 2008; Pan et al., 2010; Ganin
et al., 2016; Cortes et al., 2019). A large body of work uses
weighted-ERM based methods for domain adaptation (Ben-
david et al., 2010; Zhang et al., 2012; Blitzner et al., 2007;
Bu et al., 2022; Hanneke & Kpotufe, 2019; Redko et al.,
2017; Wang et al., 2019a; Ben-David et al., 2006); this is
either done to address domain shift or to address different
distributions of tasks in a transfer or multi-task learning
setting. This body of work is of interest for us, except that
in our case, the “source” task is actually the OOD samples.

Connection with the theory of domain adaptation
While generalization bounds for weighted-ERM like those
of Ben-David et al. (2010) are understood to be meaningful
(if not tight; see Vedantam et al. (2021)) for large sample
sizes, our work identifies an unusual non-monotonic trend
in the generalization error of the target task. Note that the
upper bound proposed by Ben-David et al. (2010) can be
used when we do not know the identity of the OOD samples
by setting $\alpha = \frac{1}{n+m}$. However, our experiments in §3.3
reveal that this bound is significantly vacuous and does
not predict the non-monotonic trends we have identified.
There is another discrepancy here, e.g., we notice that the
upper bound for naively weighted empirical error ($\alpha = 1/2$)
does not have a non-monotonic trend. A more recent paper
by Bu et al. (2022) presents an exact characterization of the
target generalization error using conditional symmetrized
Kullback-Leibler information between the output hypothesis
and target samples given the source samples. While they do
not identify non-monotonic trends in target generalization
error, their tools can potentially be useful to characterize the
phenomenon discovered in our work.

Domain Generalization seeks to learn a predictor from
multiple domains that could perform well on some unseen
test domain. This unseen test domain can be thought as
OOD data. Since no training data is available during the
training, the learner needs to make some additional assump-
tions; one popular assumption is to learn invariances across
training and testing domains (Gulrajani & Lopez-Paz, 2020;
Arjovsky et al., 2019; Sun & Saenko, 2016). We use several
benchmark datasets from this literature, but the goals of
this body of work and ours are very different because we
are interested only in generalizing on the target task, not
generalizing to the domain of the OOD samples.

Outlier and OOD Detection Identifying OOD samples
within a dataset prior to training can be thought of as a
variation of the outlier detection (OD) problem (Ben-Gal,
2010; Boukerche et al., 2020; Wang et al., 2019b; Fischler
& Bolles, 1981). These methods aim to detect outliers by
searching for the model fitted by the majority of sam-
ples. But this remains a largely unsolved problem for high-
dimensional data (Thudumu et al., 2020). Another related
but different problem is “OOD detection” (Ren et al., 2019;
Winkens et al., 2020; Fort et al., 2021; Liu et al., 2020)
which focuses on detecting data that is different from what
was used for training (also see the works of Ming et al.
(2022); Sun et al. (2022) who demonstrate that certain de-
tected OOD samples can turn out to be semantically similar
to training samples).

5. Acknowledgements

ADS and JTV were supported by the NSF AI Institute
Planning award (#2020312), NSF-Simons Research Col-
aborations on the Mathematical and Scientific Foundations
of Deep Learning (MoDL) and THEORINET. RR and PC were
supported by grants from the National Science Foundation
(IIS-2145164, CCF-2212519), Office of Naval Research
(N00014-22-1-2255), and cloud computing credits from
Amazon Web Services.
References


A. Fisher’s Linear Discriminant (FLD)

A.1. Synthetic Datasets

The target data is sampled from the distribution $P_t$ and the OOD data is sampled from the distribution $P_o$; Both distributions have two classes and one-dimensional inputs. In both distributions, each class is sampled from a univariate Gaussian distribution. The distribution of the OOD data is the target distribution translated by $\Delta$. In summary, the target distribution has the class conditional densities,

$$f_{t,0} \triangleq \mathcal{N}(-\mu, \sigma^2)$$
$$f_{t,1} \triangleq \mathcal{N}(+\mu, \sigma^2),$$

while the OOD distribution has the class conditional densities,

$$f_{o,0} \triangleq \mathcal{N}(\Delta - \mu, \sigma^2)$$
$$f_{o,1} \triangleq \mathcal{N}(\Delta + \mu, \sigma^2).$$

We also assume that both the target and OOD distributions have the same label distribution with equal class prior probabilities, i.e. $p(y_t = 1) = p(y_o = 1) = \pi = \frac{1}{2}$. Figure 1 (left) depicts $P_t$ and $P_o$ pictorially.

A.2. OOD-Agnostic Fisher’s Linear Discriminant

In this section, we derive FLD when we have samples from a single distribution – which is also applicable to the OOD-agnostic (when the identity of the OOD samples are not known) setting. Consider a binary classification problem with $D_t = \{(x_i, y_i)\}_{i=1}^n \sim P_t$ where $x_i \in X \subseteq \mathbb{R}^d$ and $y_i \in Y = \{0, 1\}$.

Let $f_k$ and $\pi_k$ be the conditional density and prior probability of class $k$ ($k \in \{0, 1\}$) respectively. The probability that $x$ belongs to class $k$ is

$$p(y = k \mid x) = \frac{\pi_k f_k(x)}{\pi_0 f_0(x) + \pi_1 f_1(x)},$$

and the maximum a posteriori estimate of the class label is

$$h(x) = \arg \max_{k \in \{0, 1\}} p(y = k \mid x) = \arg \max_{k \in \{0, 1\}} \log(\pi_k f_k(x)).$$ (3)

Fisher’s linear discriminant (FLD) assumes that each $f_k$ is a multivariate Gaussian distribution with the same covariance matrix $\Sigma$, i.e.,

$$f_k(x) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp\left(-\frac{1}{2} (x - \mu_k)^\top \Sigma^{-1} (x - \mu_k)\right).$$

Under this assumption, the joint-density $f$ of $(x, y)$ becomes,

$$f(x, y) \propto \prod_{k=0}^1 \left[ \frac{\pi_k}{|\Sigma|^{1/2}} \exp\left(-\frac{1}{2} (x - \mu_k)^\top \Sigma^{-1} (x - \mu_k)\right) \right]^{[y=k]} \text{1}_{[y=k]}$$

Figure A1. A picture of synthetic target and OOD distributions.
Therefore, the log-likelihood \( l(\mu_0, \mu_1, \Sigma, \pi_0, \pi_1) \) over \( D_t \) is given by,

\[
\ell(\mu_0, \mu_1, \Sigma, \pi_0, \pi_1) = \sum_{k=0}^1 \sum_{(x, y) \in D_{t,k}} \left[ \log \pi_k - \frac{1}{2} \log |\Sigma| - \frac{1}{2} (x - \mu_k)^\top \Sigma^{-1} (x - \mu_k) \right] + \text{const.}
\]

where \( D_{t,k} \) is the set of samples of \( D_t \) that belongs to class \( k \). Based on the likelihood function above, we can obtain the maximum likelihood estimates \( \hat{\mu}_k, \hat{\Sigma}, \hat{\pi}_k \). The expression for the estimate \( \hat{\mu}_k \) is

\[
\hat{\mu}_k = \frac{1}{|D_{t,k}|} \sum_{(x,y) \in D_{t,k}} x.
\]

Plugging these estimates into (3), we get,

\[
\hat{h}(x) = \arg \max_{k \in \{0, 1\}} \left[ \log \hat{\pi}_k - \frac{1}{2} \log |\hat{\Sigma}| - \frac{1}{2} (x - \hat{\mu}_k)^\top \hat{\Sigma}^{-1} (x - \hat{\mu}_k) \right] = \arg \max_{k \in \{0, 1\}} \left[ \log \hat{\pi}_k - \frac{1}{2} \log |\hat{\Sigma}| + x^\top \hat{\Sigma}^{-1} \hat{\mu}_k - \frac{1}{2} \hat{\mu}_k^\top \hat{\Sigma}^{-1} \hat{\mu}_k \right]
\]

Therefore, \( \hat{h}(x) = 1 \) iff,

\[
x^\top \hat{\Sigma}^{-1} \hat{\mu}_1 - \frac{1}{2} \hat{\mu}_1^\top \hat{\Sigma}^{-1} \mu_1 + \log \hat{\pi}_1 > x^\top \hat{\Sigma}^{-1} \hat{\mu}_0 - \frac{1}{2} \hat{\mu}_0^\top \hat{\Sigma}^{-1} \mu_0 + \log \hat{\pi}_0
\]

\[
x^\top \hat{\Sigma}^{-1} \hat{\mu}_1 - x^\top \hat{\Sigma}^{-1} \hat{\mu}_0 > \frac{1}{2} \hat{\mu}_1^\top \hat{\Sigma}^{-1} \mu_1 - \frac{1}{2} \hat{\mu}_0^\top \hat{\Sigma}^{-1} \mu_0 + \log \hat{\pi}_0 - \log \hat{\pi}_1
\]

\[
(\hat{\Sigma}^{-1}(\hat{\mu}_1 - \hat{\mu}_0))^\top x > (\hat{\Sigma}^{-1}(\hat{\mu}_1 - \hat{\mu}_0))^\top \left( \frac{\hat{\mu}_0 + \hat{\mu}_1}{2} \right) + \log \frac{\hat{\pi}_0}{\hat{\pi}_1}
\]

Hence the FLD decision rule \( \hat{h}(x) \) is

\[
\hat{h}(x) = \begin{cases} 
1, & \omega^\top x > c \\
0, & \text{otherwise}
\end{cases}
\]

where \( \omega = \hat{\Sigma}^{-1}(\hat{\mu}_1 - \hat{\mu}_0) \) is a projection vector and \( c = \omega^\top \left( \frac{\hat{\mu}_0 + \hat{\mu}_1}{2} \right) + \log \frac{\hat{\pi}_0}{\hat{\pi}_1} \) is a threshold. When \( d = 1 \) and \( \pi_0 = \pi_1 \), the decision rule reduces to

\[
\hat{h}(x) = \begin{cases} 
1, & x > \frac{\hat{\mu}_0 + \hat{\mu}_1}{2} \\
0, & \text{otherwise}
\end{cases}
\]  

(A.3. Deriving the Generalization Error of the Target Distribution for Synthetic Data with FLD)

We would like to derive an expression for the average generalization error of the target distribution, when we consider the synthetic data described in Appendix A.1. For simplicity, we set the variance \( \sigma^2 \) of the class conditional densities of the synthetic data to 1.

In the OOD-agnostic setting, the learning algorithm sees a single dataset \( D = D_t \cup D_o \) of size \( n + m \) which is a combination of both target and OOD samples. We can estimate \( \mu_k \) using (4) to obtain

\[
\hat{\mu}_k = \frac{1}{|D_k|} \sum_{(x,y) \in D_k} x = \frac{\sum_{(x,y) \in D_{t,k}} x + \sum_{(x,y) \in D_{o,k}} x}{n_k + m_k}
\]

\[
= \frac{n_k \bar{x}_{t,k} + m_k \bar{x}_{o,k}}{n_k + m_k} = \frac{n_k \bar{x}_{t,k} + m_k \bar{x}_{o,k}}{n + m}.
\]

where \( D_k \) is the set of samples of \( D \) that belongs to class \( k \), \( n_k = |D_{t,k}| \) and \( m_k = |D_{o,k}| \) for \( k \in \{0, 1\} \). \( \bar{x}_{t,k} \) and \( \bar{x}_{o,k} \) denote the sample means of class \( k \) in target and OOD datasets respectively. We assume that \( \pi = \frac{1}{2} \) from which it follows that \( n_k = n \pi_k = \frac{n}{2} \) and \( m_k = m \pi_k = \frac{m}{2} \). We cannot explicitly compute \( \bar{x}_{t,k} \) and \( \bar{x}_{o,k} \) when the OOD samples are not explicitly known, because we cannot separate target samples from OOD samples in \( D \).

Since the samples are drawn from Gaussians, their averages also follow Gaussian distributions. Hence, the threshold
\[ \hat{c} = \frac{\bar{\mu} + \bar{\mu}'}{2} \] of the hypothesis \( \hat{h} \), estimated using FLD, is a random variable with a Gaussian distribution i.e., \( \hat{c} \sim \mathcal{N}(\mu_h, \sigma_h^2) \) where

\[
\begin{align*}
\mu_h &= \mathbb{E}[c] = \frac{m\Delta}{n + m}, \\
\sigma_h^2 &= \text{Var}[c] = \frac{1}{n + m}.
\end{align*}
\]

The target error of a hypothesis \( \hat{h} \) is

\[
p(\hat{h}(x) \neq y | x, \hat{c}) = \frac{1}{2} p_{x \sim f_{1,1}}[x < \hat{c}] + \frac{1}{2} p_{x \sim f_{1,0}}[x > \hat{c}]
\]

\[= \frac{1}{2} \Big( 1 + \Phi(\hat{c} - \mu) - \Phi(\hat{c} + \mu) \Big) \tag{7}
\]

Using (7), the expected error on the target distribution \( \epsilon_t(\hat{h}) = \mathbb{E}_{\hat{c} \sim \mathcal{N}(\mu_h, \sigma_h^2)}[p(\hat{h}(x) \neq y | x, \hat{c})] \) is given by,

\[
\epsilon_t(\hat{h}) = \int_{-\infty}^{\infty} \frac{1}{2} \Big[ 1 + \Phi(\hat{c} - \mu) - \Phi(\hat{c} + \mu) \Big] \frac{1}{\sigma_h} \phi\left( \frac{\hat{c} - \mu_h}{\sigma_h} \right) d\hat{c}
\]

\[= \int_{-\infty}^{\infty} \frac{1}{2} \Big[ 1 + \Phi(y\sigma_h + \mu_h - \mu) - \Phi(y\sigma_h + \mu_h + \mu) \Big] \phi(y) dy
\]

\[= \frac{1}{2} \left[ \Phi\left( \frac{\mu_h - \mu}{\sqrt{1 + \sigma_h^2}} \right) + \Phi\left( \frac{-\mu_h - \mu}{\sqrt{1 + \sigma_h^2}} \right) \right]
\]

In the last equality, we make use of the identity \( \int_{-\infty}^{\infty} \Phi(cx + d)\phi(x)dx = \Phi\left( \frac{d}{\sqrt{c^2 + 1}} \right) \) where \( \phi \) and \( \Phi \) are the PDF and CDF of the standard normal. Substituting the expressions for \( \mu_h, \sigma_h^2 \) into the above equation, we get

\[
\epsilon_t(\hat{h}) = \frac{1}{2} \left[ \Phi\left( \frac{-m\Delta - (n + m)\mu}{\sqrt{(n + m)(n + m + 1)}} \right) + \Phi\left( \frac{-m\Delta - (n + m)\mu}{\sqrt{(n + m)(n + m + 1)}} \right) \right] \tag{8}
\]

For synthetic data with \( \sigma^2 \neq 1 \), the target generalization error can be obtained by simply replacing \( \mu \) and \( \Delta \) with \( \frac{\mu}{\sigma} \) and \( \frac{\Delta}{\sigma} \) respectively in (8).

\textbf{A.4. OOD-Aware Weighted Fisher’s Linear Discriminant}

We consider a target dataset \( D_t = \{(x_i, y_i)\}_{i=1}^n \) and an OOD dataset \( D_o = \{(x_i, y_i)\}_{i=1}^m \), which are samples from the synthetic data from Appendix A.1. This setting differs from Appendix A.3 since we know whether each sample from \( D = D_t \cup D_o \) is OOD or not. This difference allows us to consider a log-likelihood function that weights the target and OOD samples differently, i.e. we consider

\[
\ell(\mu_0, \mu_1, \sigma_0^2, \sigma_1^2) = \sum_{k=0}^{1} \alpha \sum_{(x,y) \in D_{t,k}} \left\{ -\log \sigma_k - \frac{(x - \mu_k)^2}{2\sigma_k^2} \right\} + (1 - \alpha) \sum_{(x,y) \in D_{o,k}} \left\{ -\log \sigma_k - \frac{(x - \mu_k)^2}{2\sigma_k^2} \right\} + \text{const.} \tag{9}
\]

\( \alpha \) is a weight that controls the contribution of the OOD samples in the log-likelihood function. Under the above log-likelihood, the maximum likelihood estimate for \( \mu_k \) is

\[
\hat{\mu}_k = \frac{\alpha \sum_{(x,y) \in D_{t,k}} x + (1 - \alpha) \sum_{(x,y) \in D_{o,k}} x}{\alpha |D_{t,k}| + (1 - \alpha)|D_{o,k}|}. \tag{10}
\]

We can make use of the above \( \hat{\mu}_k \) to get a weighted FLD decision rule using (5).

\textbf{A.5. Deriving the Generalization Error of the Target Distribution for Synthetic Data with Weighted FLD}

We consider the synthetic distributions in Appendix A.1 with \( \sigma^2 = 1 \). We re-write \( \hat{\mu}_k \) from (10) using notation from Appendix A.3:

\[
\hat{\mu}_k = \frac{n\alpha \bar{x}_{t,k} + m(1 - \alpha)\bar{x}_{o,k}}{n\alpha + m(1 - \alpha)}.
\]
We can explicitly compute \( \bar{x}_{t,k} \) and \( \bar{x}_{o,k} \) in the OOD-aware setting since we can separate target samples from OOD samples. For the synthetic distribution, the threshold \( \hat{c}_{\alpha} = \frac{\hat{\mu} + \hat{\sigma}}{2} \) of the hypothesis \( \hat{h}_{\alpha} \) follows a normal distribution \( \mathcal{N}(\mu_{h\alpha}, \sigma_{h\alpha}^2) \) where

\[
\mu_{h\alpha} = \mathbb{E}[c_{\alpha}] = \frac{m(1 - \alpha)\Delta}{m\alpha + m(1 - \alpha)}
\]

\[
\sigma_{h\alpha}^2 = \text{Var}[c_{\alpha}] = \frac{\alpha^2 n + (1 - \alpha)^2 m}{(m\alpha + m(1 - \alpha))^2}
\]

Similar to the Appendix A.3, we derive an analytical expression for the expected target risk of the weighted FLD, which is

\[
e_t(\hat{h}_{\alpha}) = \frac{1}{2} \left[ \Phi \left( \frac{\mu_{h\alpha} - \mu}{\sqrt{1 + \sigma_{h\alpha}^2}} \right) + \Phi \left( \frac{-\mu_{h\alpha} - \mu}{\sqrt{1 + \sigma_{h\alpha}^2}} \right) \right]
\]

(A.6. Additional Experiments using FLD)

![Figure A2](https://example.com/figure2.png)

Figure A2. The FLD generalization error (Y-axis) on the target distribution is plotted against the ratio of OOD samples to target samples (X-axis). Figures (a) and (c) are plotted using the analytical expressions in (8) and (11) respectively while figures (b) and (d) are the corresponding plots from Monte-carlo simulations. The Monte-carlo simulations agree with the plots from the analytical expression, which validates its correctness. (a) and (b): The figure is identical to Figure 1 and considers synthetic data with \( n = 100, \mu = 5 \) and \( \sigma = 10 \) in the OOD-agnostic setting. While a small number of OOD samples improves generalization on the target distribution, lots of samples increase the generalization error on the target distribution. (c) and (d): The figures consider synthetic data with \( n = 4, \mu = 1 \) and \( \sigma = 1 \) in the OOD-aware setting. If we consider the weighted FLD trained with optimal \( \alpha^* \), then the average generalization error monotonically decreases with more OOD samples. Shaded regions indicate 95% confidence intervals over the Monte-Carlo replicates.

A.7. Deriving the Upper Bound in Theorem 3 for the OOD-Agnostic Fisher’s Linear Discriminant

We begin by defining the following quantities: Given a hypothesis \( h : X \to \{0, 1\} \), the probability according to the distribution \( P_t \) that \( h \) disagrees with a labeling function \( f \) is defined as,

\[
e_s(h, f) = \mathbb{E}_{x \sim P_t}[|h(x) - f(x)|]
\]

For a hypothesis space \( H \), (Ben-David et al., 2010) defines the divergence measure between two distributions \( P_t \) and \( P_o \) in the symmetric difference hypothesis space as,

\[
d_H^s(P_t, P_o) = 2 \sup_{h, h' \in H} |e_s(h, h') - e_t(h, h')|
\]

With these definitions in place, we restate a slightly modified version of the Theorem 3 from (Ben-David et al., 2010) below.
Theorem 4. Let \( H \) be a hypothesis space of VC dimension \( d \). Let \( D \) be a dataset generated by drawing \( n \) samples from a target distribution \( P_t \) and \( m \) OOD samples from \( P_o \). If \( \hat{h} \in H \) is the empirical minimizer of \( \alpha e_t(h) + (1 - \alpha)e_o(h) \) on \( D \) and \( h^* = \min_{h \in H} e_t(h) \) is the target error minimizer, then for any \( \delta \in (0, 1) \), with probability at least \( 1 - 1 \) (over the choice of samples),

\[
e_t(\hat{h}) \leq e_t(h^*) + 4\sqrt{\frac{\alpha^2}{n} + \frac{(1 - \alpha)^2}{m}} \sqrt{2d \log(2(n + m + 1)) + 2 \log \left( \frac{8}{\delta} \right) + 2(1 - \alpha)\left( \frac{1}{2}d_H(P_t, P_o) + \lambda \right)}
\]

(12)

where, \( \lambda \) is the combined error of the ideal joint hypothesis given by \( h^* = \arg\min_{h \in H} e_t(h) + e_o(h) \). Hence, \( \lambda = e_t(h^*) + e_o(h^*) \).

We wish to adapt the above theorem according to our FLD example in §2.1 and consequently find an expression for the upper bound \( U(n, m, d_H(P_t, P_o)) \) in terms of \( n, m \) and \( \Delta \). As we do not know of the existence of OOD samples in dataset \( D \), we find the hypothesis \( \hat{h} \) by minimizing the empirical loss below.

\[
\hat{e}(h) = \frac{1}{n + m} \sum_{i=1}^{n+m} \ell(h(x_i), y_i) = \frac{1}{n + m} \sum_{(x, y) \in D_t} \ell(h(x), y) + \frac{1}{n + m} \sum_{(x, y) \in D_o} \ell(h(x), y) = \frac{n}{n + m} e_t(h) + \frac{m}{n + m} e_o(h).
\]

Here, we have assumed that \( \ell(\cdot) \) is the 0-1 loss. Therefore, under the OOD agnostic setting, we minimize the objective function \( e(h) = \alpha e_t(h) + (1 - \alpha)e_o(h) \) where \( \alpha = n/(n + m) \). Since we deal with a univariate FLD, the VC dimension of the hypothesis space is equal to \( d = 1 + 1 = 2 \). Plugging these terms in (12), we can rewrite the upper bound as,

\[
U(n, m, d_H(P_t, P_o)) = e_t(h^*) + 4\sqrt{4 \log(2(n + m + 1)) + 2 \log \left( \frac{8}{\delta} \right) + \frac{2m}{n + m} \left( \frac{1}{2}d_H(P_t, P_o) + \lambda \right)}
\]

(13)

The first term of the above expression corresponds to the error of the best hypothesis \( h^*_t \) in class \( H \) for the target distribution \( P_t \). Thus, \( e_t(h^*_t) \) is equivalent to the Bayes optimal error or the lowest possible error achievable for the target distribution, under \( H \). By setting \( m = 0 \) in (8), we arrive at the expected error \( e_t(\hat{h}) \) on the target distribution when we estimate \( \hat{h} \) using \( n \) target samples. The Bayes optimal error \( e_t(h^*_t) \) is then equal to the limit of \( e_t(\hat{h}) \) as \( n \to \infty \).

\[
e_t(h^*_t) = \lim_{n \to \infty} e_t(\hat{h}) = \lim_{n \to \infty} \Phi \left( -\frac{n(\mu/\sigma)}{\sqrt{n(n + 1)}} \right) = \Phi(-\mu/\sigma)
\]

Intuitively, the threshold corresponding to the ideal joint hypothesis \( h^* \) for our FLD example is given by the mid point between the centers of the two distributions,

\[
h^*(x) = \arg\min_{h \in H} e_o(h) + e_t(h) = \frac{\Delta}{2} \mathbb{1}_{(\Delta/2, \infty)}(x)
\]

where \( I_A(x) \) is the indicator function of the subset \( A \). Therefore, the combined error \( \lambda \) of the ideal joint hypothesis can be computed as follows.

\[
\lambda = e_o(h^*) + e_t(h^*) = \frac{1}{2} p_{x \sim f_{t,0}}[x > \Delta/2] + \frac{1}{2} p_{x \sim f_{t,1}}[x < \Delta/2] + \frac{1}{2} p_{x \sim f_{o,0}}[x > \Delta/2] + \frac{1}{2} p_{x \sim f_{o,1}}[x < \Delta/2] = \Phi\left(-\frac{\Delta/2 - \mu}{\sigma}\right) + \Phi\left(\frac{\Delta/2 - \mu}{\sigma}\right)
\]

Finally, we turn to the divergence term \( d_H(P_t, P_o) \). Let \( h, h' \in H \) be two hypotheses with thresholds \( \epsilon \) and \( \epsilon' \), respectively.
From the definition of $e_t(h, h')$ we have,

$$e_t(h, h') = \mathbb{E}_t \left[ |h(x) - h'(x)| \right]$$

$$= \mathbb{E}_t \left[ \mathbb{1}_{(c, \infty)}(x) - \mathbb{1}_{(c', \infty)}(x) \right]$$

$$= \mathbb{E}_t \left[ \mathbb{1}_{\{\min(c, c'), \max(c, c')\}}(x) \right]$$

$$= \mu_t[\min(c, c') < x \leq \max(c, c')]$$

$$= \frac{1}{2} [\Phi(\frac{\max(c, c') + \mu}{\sigma}) + \Phi(\frac{\min(c, c') - \mu}{\sigma})] - \Phi(\frac{\min(c, c') + \mu}{\sigma}) - \Phi\left(\frac{\max(c, c') - \mu}{\sigma}\right)$$

Similarly, we can show that $e_0(h, h') = \psi_{\mu, \sigma}(c - \Delta, c' - \Delta)$. Therefore, we can rewrite the expression for $d_H(P_t, P_o)$ as follows.

$$d_H(P_t, P_o) = 2 \sup_{h, h' \in H} |e_0(h, h') - e_t(h, h')| = 2 \sup_{c, c' \in [0, \Delta]} |\psi_{\mu, \sigma}(c - \Delta, c' - \Delta) - \psi_{\mu, \sigma}(c, c')| = d_H^*(\Delta)$$

Using this expression we can numerically compute $d_H^*$, given the values of $\mu, \sigma$ and $\Delta$. Plugging in the expressions we have obtained for $e_t(h^*_n), \lambda$ and $d_H(P_t, P_o)$ in (13), we arrive at the desired upper bound for the expected target error $e_t(\hat{h})$ of our FLD example.

$$U(n, m, \Delta) = \Phi(-\mu/\sigma) + 4\sqrt{4\log(2(n + m + 1)) + 2\log\left(\frac{8}{\delta}\right)} + \frac{2m}{n + m} \left[ \frac{1}{2} d_H^*(\Delta) + \Phi\left(\frac{-\Delta/2 - \mu}{\sigma}\right) + \Phi\left(\frac{\Delta/2 - \mu}{\sigma}\right) \right]$$

(A.8. Comparisons between the Upper Bound and the True Target Generalization Error)

---

**Figure A3.** The upper bound (as computed by (14)) and the true expected target error (as computed by (8)), for 3 different variations of the FLD example in §2.1. In the left and right columns, we observe that the shape of the curve agrees somewhat with that of the true error. Notice that the separation $\Delta$ between the distributions of the target and OOD data is large in these cases. Figure 12 and the middle column of the current figure indicate that the upper bound does not exhibit a non-monotonic trend while the true error does. It is also important to note that the bound is significantly vacuous in all cases. These observations suggest that the Theorem 3 from the work of Ben-David et al. (2010) does not explain the non-monotonic trends that we have identified in this work.

---

**B. Experiments with Neural Networks**

**B.1. Datasets**

We experiment on images from CIFAR-10, CIFIC-10 (Darlow et al., 2018) and several datasets from the DomainBed benchmark (Gulrajani & Lopez-Paz, 2020): Rotated MNIST (Ghifary et al., 2015), PACS (Li et al., 2017), and Domain-Net (Peng et al., 2019). We construct sub-tasks from these datasets as explained below.
CIFAR-10  We use of tasks from Split-CIFAR10 (Zenke et al., 2017) which are five binary classification sub-tasks constructed by grouping consecutive labels of CIFAR-10. The 5 task distributions are airplane vs. automobile (T_1), bird vs. cat (T_2), deer vs. dog (T_3), frog vs. horse (T_4) and ship vs truck (T_5). All the images are of size (3, 32, 32).

CINIC-10  This dataset combines CIFAR-10 with downscaled images from ImageNet. It contains images of size (3, 32, 32) across 10 classes (same classes as CIFAR-10). As there are two sources of the images within this dataset, it is a natural candidate for studying distribution shift. The construction of the dataset motivates us to consider two distributions from CINIC-10: (1) Distribution with only CIFAR images, and (2) Distribution with only ImageNet images.

Rotated MNIST  This dataset is constructed from MNIST by rotating the images (which are of size (1, 28, 28)). All MNIST images rotated by an angle $\theta^\circ$ are considered to belong to the same distribution. Hence, we can consider the family of distributions which is characterized by 10-way classification of hand-written digit images rotated $\theta^\circ$. By varying $\theta$, we can obtain a number of different distributions.

PACS  PACS contains images of size (3, 224, 244) with 7 classes present across 4 domains {art, cartoons, photos, sketches}. In our experiments, we consider only 3 classes ({Dog, Elephant, Horse}) out of the 7 and consider the 3-way classification of images from a given domain as a distribution. Therefore, we can have a total of 4 distinct distributions from PACS.

DomainNet  Similar to PACS, this dataset contains images of size (3, 224, 244) from 6 domains {clipart, infograph, painting, quickdraw, real, sketches} across 345 classes. In our experiments, we consider only 2 classes, (Bird, Plane) and consider the binary classification of images from a given domain as a distribution. As a result, we can have a total of 6 distinct distributions from PACS.

B.2. Forming Target and OOD Distributions

We consider two types of setups to study the impact of OOD data:

OOD data arising due to geometric intra-class nuisances  We study the effect of intra-class nuisances using a classification task using samples from a target distribution and OOD samples from a transformed version of the same distribution. In this regard, we consider the following experimental setups.

1. Rotated MNIST: unrotated images as target and $\theta^\circ$-rotated images as OOD: We consider the 10-way classification (see Appendix B.1) of unrotated images as the target data and that of the $\theta^\circ$-rotated images as the OOD data. We can have different OOD data by selecting different values for $\theta$.

2. Rotated CIFAR-10: $T_2$ as target and rotated $T_2$ as OOD: We choose the bird vs. cat ($T_2$) task from Split-CIFAR10 as the target distribution. We then rotate the images of $T_2$ by an angle $\theta^\circ$ counter-clockwise around their centers to form a new task distribution denoted by $\theta$-$T_2$, which we consider as OOD. Different OOD datasets can be obtained by selecting different values for $\theta$.

3. Blurred CIFAR-10: $T_4$ as target and blurred $T_4$ as OOD: We choose the Frog vs. Horse ($T_4$) task from Split-CIFAR10 as the target distribution. We then add Gaussian blur with standard deviation $\sigma$ to the images of $T_4$ to form a new task distribution denoted by $\sigma$-$T_2$, which we consider as the OOD. By setting distinct values for $\sigma$, we have different OOD datasets.

OOD data arising due to category shifts and concept drifts  We study this aspect using two different target and OOD classification problems as described below.

1. Split-CIFAR10: $T_1$ as Target and $T_1$ as OOD: We choose a pair of distinct tasks from the 5 binary classification tasks of Split-CIFAR10 and consider one as the target distribution and the other as the OOD. We perform experiments for all pairs of distributions (20 in total) in Split-CIFAR10.

2. PACS: Photo-domain as target and X-domain as OOD: Out of the four 3-way classification tasks from PACS described in Appendix B.1, we select the photo-domain as the target distribution and consider one of the remaining 3 domains (for instance, the sketch-domain) as the OOD.

3. DomainNet: Real-domain as target and X-domain as OOD: Out of the six binary classification tasks from DomainNet described in Appendix B.1, we consider the real-domain as the target distribution and select one of the remaining 5 domains (for instance, the painting-domain) as OOD.

4. CINIC-10: CIFAR10 as target and ImageNet as OOD: Here we simply select the 10-way classification of CIFAR images as the target distribution and that of ImageNet as OOD.
B.3. Experimental Details

In the above experiments, for each random seed, we randomly select a fixed sample of size \( n \) from the target distribution. Next, we select OOD samples of varying sizes \( m \) such that the previous samples are a subset of the next set of samples. The samples from both target and OOD distributions preserve the ratio of the classes. For rotated MNIST, rotated CIFAR-10, and blurred CIFAR-10, when selecting multiple sets of OOD samples, the OOD images that correspond to the \( n \) selected target images are disregarded. For PACS and DomainNet, the images are downsampled to \((3, 64, 64)\) during training.

For both the OOD-agnostic (OOD unknown) and OOD-aware (OOD known) settings, at each \( m \)-value, we construct a combined dataset containing the \( n \) sized target set and \( m \) sized OOD set. We use a CNN (see Appendix B.4) for experiments in the both of these settings. We experiment with \( \alpha \) fixed to 0.5 (naive OOD-aware model) and with the optimal \( \alpha^* \). We average the runs over 10 random seeds and evaluate on a test set comprised of only target samples.

In the optimal OOD-aware setting, we use a grid-search to find the optimal \( \alpha^* \) for each value of \( m \). We use an adaptive equally-spaced \( \alpha \) search set of size 10 such that it ranges from \( \alpha^*_\text{prev} \) to 1.0 (excluding 1.0) where \( \alpha^*_\text{prev} \) is the optimal value of \( \alpha \) corresponding to the previous value of \( m \). We use this search space since we expect \( \alpha^* \) to be an increasing function of \( m \).

B.4. Neural Architectures and Training

We primarily use 3 different network architectures in our experiments: (a) a small convolutional network with 0.12M parameters (denoted by SmallConv), (b) a wide residual network (Zagoruyko & Komodakis, 2016) of depth 10 and widening factor 2 (WRN-10-2), and (c) a larger wide residual network of depth 16 and widening factor 4 (WRN-16-4). SmallConv comprises of 3 convolution layers (kernel size 3 and 80 filters) interleaved with max-pooling, ReLU, batch-norm layers, with a fully-connected classifier layer in our experiments.

Table A1 provides a summary of network architectures used in the experiments described earlier. All the networks are trained using stochastic gradient descent (SGD) with Nesterov’s momentum and cosine-annealed learning rate. The hyperparameters used for the training are, learning rate of 0.01, and a weight-decay of \( 10^{-5} \). All the images are normalized to have mean 0.5 and standard deviation 0.25. In the OOD-agnostic setting, we use sampling without replacement to construct the mini-batches. In the OOD-aware settings (both naive and optimal), we construct mini-batches with a fixed ratio of target and OOD samples. See Appendix B.5 and Figure A4 for more details.

<table>
<thead>
<tr>
<th>Experiment</th>
<th>Network(s)</th>
<th># classes</th>
<th>( n )</th>
<th>Image Size</th>
<th>Mini-Batch Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rotated MNIST</td>
<td>SmallConv</td>
<td>10</td>
<td>100</td>
<td>(1,28,28)</td>
<td>128</td>
</tr>
<tr>
<td>Rotated CIFAR-10</td>
<td>SmallConv, WRN-10-2</td>
<td>2</td>
<td>100</td>
<td>(3,32,32)</td>
<td>128</td>
</tr>
<tr>
<td>Blurred CIFAR-10</td>
<td>WRN-10-2</td>
<td>2</td>
<td>100</td>
<td>(3,32,32)</td>
<td>128</td>
</tr>
<tr>
<td>Split-CIFAR10</td>
<td>SmallConv, WRN-10-2</td>
<td>2</td>
<td>100</td>
<td>(3,32,32)</td>
<td>128</td>
</tr>
<tr>
<td>PACS</td>
<td>WRN-16-4</td>
<td>3</td>
<td>30</td>
<td>(3,64,64)</td>
<td>16</td>
</tr>
<tr>
<td>DomainNet</td>
<td>WRN-16-4</td>
<td>2</td>
<td>50</td>
<td>(3,64,64)</td>
<td>16</td>
</tr>
<tr>
<td>CINIC-10</td>
<td>WRN-10-2</td>
<td>10</td>
<td>100</td>
<td>(3,32,32)</td>
<td>128</td>
</tr>
</tbody>
</table>

Table A1. Summary of network architectures used in the experiments

B.5. Construction of Mini-Batches

Consider a mini-batch \( \{(x_b, y_b)\}_{b=1}^B \) of size \( B \). Let the randomly chosen mini-batch contains \( B_t \) target samples and \( B_o \) OOD samples \((B = B_t + B_o)\). Let \( \hat{e}_{B,t}(h) \) and \( \hat{e}_{B,o}(h) \) denote the average mini-batch surrogate losses for the \( B_t \) target samples and \( B_o \) OOD samples respectively.

In the OOD-aware (when we know which samples are OOD) setting, \( \hat{e}_{B,t}(h) \) and \( \hat{e}_{B,o}(h) \) can be computed explicitly for each mini-batch resulting in the mini-batch gradient

\[
\nabla \hat{e}_B(h) = \alpha \nabla \hat{e}_{B,t}(h) + (1 - \alpha) \nabla \hat{e}_{B,o}(h).
\]

If we were to sample without replacement, we expect the fraction of the target samples in every mini-batch to approximately equal \( \frac{n}{n + m} \) on average. However, if \( m >> n \), we run into a couple of issues. First, we observe that most mini-batches have no target samples, making it impossible to compute \( \nabla \hat{e}_{B,t}(h) \). Next, even if the mini-batch does have some target samples, there are very few of them, resulting in high variance in the estimate \( \nabla \hat{e}_{B,t}(h) \).

Hence, we find it beneficial to consider alternative sampling schemes for the mini-batch. Independent of the values of \( n \) and \( m \),
and $m$, we use a sampler which ensures that every mini-batch has a fixed fraction of target samples, which we denote by $\beta$. For example if the mini-batch size $B$ is 20 and if $\beta = 0.5$, then every mini-batch has 10 target samples and 10 OOD samples regardless of $n$ and $m$. Note that this sampling biases the gradient, but results in reduced variance estimates. In practice, we observe improved test errors when we set $\beta$ to either 0.5 or 0.75.

![Figure A4. Standard mini-batching strategy versus ensuring that every mini-batch has a fraction $\beta$ samples from the target distribution.](image)

Figure A4. **Standard mini-batching strategy versus ensuring that every mini-batch has a fraction $\beta$ samples from the target distribution.** The test error of a neural network (SmallConv) on the target distribution (Y-axis) is plotted against the number of OOD samples (X-axis) for the target-OOD pair of $T_1$ and $T_5$. One set of curves (lightest shade of green and yellow) considers mini-batches which are constructed using sampling without replacement; This is the standard strategy used in supervised learning. The other curves consider $\beta = 0.5$ (intermediate shades of orange and green) and $\beta = 0.75$ (darkest shade of red and green). All plots are in the OOD-aware setting. **Left:** If we consider $\alpha = 0.5$, then the choice of $\beta$ has little effect on the generalization error. **Right:** However, if we use $\alpha^*$ to weight the OOD and target losses, then the generalization error depends on the the choice of $\beta$ with $\beta = 0.75$ having the lowest test error.

**B.6. Additional Experiments with Neural Networks**

![Figure A5.](image)

Figure A5. We plot the generalization error on the target distribution (Y-axis) against the number of OOD samples $m$ (X-axis) across three different target sample sizes, $n = 50$, 100 and 200 for the target-OOD pair $T_2$ and $T_5$ from Split-CIFAR10. Non-monotonic trends in generalization error are present in all the three cases. The trend is less apparent for $n = 50$ since the number of samples is small resulting in a large variance. Error bars indicate 95% confidence intervals (10 runs).
Figure A6. (a) We plot the test error of SmallConv on the target distribution (Y-axis) against the ratio of number of OOD samples to the number of samples from the target task (X-axis), for all target-OOD pairs from Split-CIFAR10. A neural net trained with a loss weighted by $\alpha^*$ is able to leverage OOD data to improve the networks ability to generalize on the target distribution. Shaded regions indicate 95% confidence intervals over 10 experiments. (b) The optimal $\alpha^*$ (Y-axis) is plotted against the number of OOD samples (X-axis) for the optimally weighted OOD-aware setting. As we increase the number of OOD samples, we see that $\alpha^*$ increases. This allows us to balance the variance from few target samples and the bias from using OOD samples from a different distribution.
Figure A7. (a) We plot the test error of WRN-10-2 on the target distribution (Y-axis) against the ratio of number of OOD samples to the number of samples on the target task (X-axis), for all target-OOD pairs from Split-CIFAR10. A neural net trained with a loss weighted by $\alpha^*$ is able to leverage OOD data to improve the network’s ability to generalize on the target distribution. Shaded regions indicate 95% confidence intervals over 10 experiments. (b) The optimal $\alpha^*$ (Y-axis) is plotted against the number of OOD samples (X-axis) for the optimally weighted OOD-aware setting. As we increase the number of OOD samples, we see that $\alpha^*$ increases. This allows us to balance the variance from few target samples and the bias from using OOD samples from a different distribution.
Figure A8. **Left:** A binary classification problem (Bird vs. Cat) is the target distribution and images of these classes rotated by different angles $\theta$ are OOD. We see non-monotonic curves for larger values of $\theta$. For $135^\circ$ in particular, the generalization error at $m/n = 50$ is worse than the generalization error with no OOD samples, i.e. OOD samples actively hurt generalization.

**Middle:** Generalization error on the target distribution is plotted against the number of OOD samples for 3 different target-OOD pairs constructed from CIFAR-10 for three settings: OOD-agnostic ERM where we minimize the total average risk over both distributions (red), an objective which minimizes the sum of the average loss of the target and OOD distributions which corresponds to $\alpha = 1/2$ (OOD-aware, yellow) and an objective which minimizes an optimally weighted convex combination of the target and OOD empirical loss (green).

**Right:** The optimal $\alpha^*$ obtained via grid search for the three problems in the middle column plotted against different number of OOD samples. Note that the appropriate value of $\alpha$ lies very close to 1 but it is never exactly 1. In other words the OOD samples always benefit if we use the weighted objective in Theorem 3, even if this benefit is marginal in cases when OOD samples are very different from those of the target.

Figure A9. We consider a 40-class classification problem from DomainNet where the classes are animals from three super-classes: mammals, cold blooded animals and birds. The target distribution considers images of animals from the “real” domain. OOD data considers images from the domains “paintings”, “quickdraw” and “sketches”. We plot the target generalization error against the ratio of OOD and target samples and observe the risk to be non-monotonic for 2 of the 3 OOD domains. Note that the error of the trained network (0.85) is lower than the error of a classifier that predicts all classes with uniform probability (0.975). The error is high because we use very few training samples; the number of target samples is 200 (i.e. only 5 samples per class). Note that the error bars indicate 95% confidence intervals over 3 runs.