IMPROVING OUT-OF-DISTRIBUTION ROBUSTNESS VIA SELECTIVE AUGMENTATION

Anonymous authors
Paper under double-blind review

Abstract

Machine learning algorithms typically assume that training and test examples are drawn from the same distribution. However, distribution shifts is a common problem in real-world applications and can cause models to perform dramatically worse at test time. In this paper, we specifically consider the problems of domain shifts and subpopulation shifts, where learning invariant representations by aligning domain-specific representations or balancing the risks across domains with regularizers are popular solutions. However, designing regularizers that are suitable for diverse real-world datasets is challenging. Instead, we shed new light on addressing distribution shifts by directly eliminating domain-related spurious correlations with augmentation, leading to a simple technique based on mixup, called LISA (Learning Invariant Representations via Selective Augmentation). LISA selectively interpolates samples either with the same labels but different domains or with the same domain but different labels. Empirically, we study the effectiveness of LISA on nine benchmarks ranging from subpopulation shifts to domain shifts. The results indicate that LISA consistently outperforms other state-of-the-art methods with superior invariant representations. The empirical findings are further strengthened by our theoretical analysis.

1 Introduction

To deploy machine learning algorithms in real-world applications, we must pay attention to distribution shifts, i.e., when the test distribution is different from the training distribution, which substantially degrades model performance. We will consider performance gaps caused by two kinds of distribution shifts: domain shifts and subpopulation shifts. In domain shifts, the test data is sampled from different domains than the training data, which requires the trained model to generalize well to test domains without seeing training data from those domains. Take health risk prediction as an example. We may want to train a model on patients from a few sampled hospitals and then deploy the model to a broader set of hospitals [Koh et al., 2021]. In subpopulation shifts, the proportions of subpopulations in the test distribution differ from the proportions in the training distribution. When subpopulation shift occur, models perform poorly when they falsely rely on spurious correlations, which may occur when some subpopulations are under-represented in the training set. For example, in financial risk prediction, a machine learning model trained on the entire population may associate the labels with demographic features (e.g., religion and race), making the model fail on the test set when such an association does not hold in reality.

To improve model robustness under these two kinds of distribution shifts, methods for learning invariant representations have shown effectiveness in various applications. These methods learn features or prediction mechanisms that are invariant to different domains while still containing sufficient information for the targeted task [Li et al., 2018; Arjovsky et al., 2019]. Concretely, some prior works learn invariant representations by aligning and regularizing the domain-specific representations [Li et al., 2018; Sun & Saenko, 2016]. Other works aim to find invariant representations by balancing the risk across domains using regularizers [Arjovsky et al., 2019; Krueger et al., 2021; Rosenfeld et al., 2021], which further increases the dependency between the invariant representations and labels. However, designing regularizers that are widely suitable to datasets from diverse domains is especially challenging and the regularizers themselves may also adversely limit the model capacity, leading to inconsistent performance among various real-world datasets. For example, on the WILDS datasets [Koh et al., 2021], invariant risk minimization (IRM) [Arjovsky
et al., 2019} outperforms empirical risk minimization (ERM) on CivilComments, but fails to improve robustness on a variety of other datasets like Camelyon17 and RxRx1. A similar phenomenon is also reflected in the performance of CORAL (Sun & Saenko, 2016).

Instead of explicitly imposing regularization to learn invariant representations, we turn towards an implicit solution. Inspired by mixup (Zhang et al., 2018), we aim to alleviate the effects of domain-related spurious information through data interpolation, leading to a simple algorithm called LISA (Learning Invariant Representations with Selective Augmentation). Concretely, LISA linearly interpolates the features for a pair of samples and applies the same interpolation strategy on the corresponding labels. Crucially, the pairs are selectively chosen according to two sample selection strategies. In selection strategy I, LISA interpolates samples with the same label but different domains, where the model should ignore the domain information and generate different predicted values as the interpolation ratio changes. In this way, LISA encourages the model to learn domain-invariant predictors without explicitly constraining or regularizing the representation.

The primary contributions of this paper are as follows: (1) We develop a method that tackles the problem of distribution shifts by canceling out the domain-related spurious correlations via data interpolation. (2) We conduct broad empirical experiments to evaluate the effectiveness of LISA on nine benchmark datasets from diverse domains. In these experiments, we make the following observations. First, we find that LISA consistently outperforms seven prior methods in addressing both domain shifts and subpopulation shifts. Second, we identify that the performance gains of LISA are indeed caused by canceling out domain-specific information and learning invariant representations, rather than simply involving more data via interpolation. Third, when the degree of distribution shift increases, LISA achieves more significant performance gains. (3) Finally, we provide theoretical analysis of the phenomena distilled from the empirical studies, where we provably demonstrate that LISA can achieve smaller worst-domain error compared with ERM and vanilla mixup. We also note that to the best of our knowledge, this is the first theoretical analysis of how mixup (with or without the selection strategies) affects mis-classification error.

### 2 Preliminaries

In this paper, we consider the setting where one predicts the label $y \in \mathcal{Y}$ based on the input feature $x \in \mathcal{X}$. Given a parameter space $\Theta$ and a loss function $\ell$, we are supposed to train a model $f_\theta$ under the training distribution $P_{tr}$, where $\theta \in \Theta$. In empirical risk minimization (ERM), assume the empirical distribution over training data is $\hat{P}_{tr}$, ERM optimizes the following objective:

$$\theta^* := \arg\min_{\theta \in \Theta} \mathbb{E}_{(x,y) \sim \hat{P}}[\ell(f_\theta(x), y)].$$

In a traditional machine learning setting, a test set, sampled from a test distribution $P_{ts}$, is used to evaluate the generalization of the trained model $\theta^*$, where the test distribution is assumed to be the same as the training distribution, i.e., $P_{tr} = P_{ts}$. In this paper, we are interested in the setting when distribution shift occurs, i.e., $P_{tr} \neq P_{ts}$.

Specifically, follow Koh et al. (2021), we regard the overall data distribution containing $D = \{1, \ldots, D\}$ domains and each domain $d \in D$ is associated with a data distribution $P_d$ over a set $(X, Y, d) = \{(x_i, y_i, d)\}_{i=1}^{N_d}$, where $N_d$ is the number of samples in domain $d$. Then, we formulate the training distribution as the mixture of $D$ domains, i.e., $P_{tr} = \sum_{d \in D} \alpha_d r_d^{tr} P_d$, where $\{r_d^{tr}\}$ denotes the mixture probabilities in training set. Here, the training domains are defined as $D^{tr} = \{d \in D | r_d^{tr} > 0\}$. Similarly, the test distribution could be represented as $P_{ts} = \sum_{d \in D} \alpha_d r_d^{ts} P_d$, where $\{r_d^{ts}\}$ is the mixture probabilities in test set. The test domains are defined as $D^{ts} = \{d \in D | r_d^{ts} > 0\}$.

![Figure 1: Group illustration of Waterbird data. Domains and labels are represented by background and bird type. Most training samples are drawn from waterbird in water ($d_1, y_1$) and landbird in land ($d_2, y_2$). The trained model is spuriously biased by the background.](image-url)
In domain shifts, we investigate the problem that the test domains are disjoint from the training domains, i.e., $D^t \cap D^s = \emptyset$. In general, we assume the test domains share some common properties with the training domains. For example, in Camelyon17 (Koh et al., 2021) data, we train the model on some hospitals and test it in a new hospital. We evaluate the worst-domain or average performance of the classifier among all test domains.

In subpopulation shifts, the test set have domains that have been seen in the training set, but with a different proportion of subpopulations, i.e., $D^s \subseteq D^t$ but $\{r^s_d\} \neq \{r^t_d\}$. Under this setting, follow (Sagawa et al., 2020a), we specially consider group-based spurious correlations, where each group $g \in G$ is defined to be associated with a domain $d$ and a label $y$, i.e., $g = (d, y)$. We assume the domain identification is spuriously correlated with the label. For example, we illustrate the Waterbirds dataset in Figure 1, where the background $d$ (water or land) is spuriously correlated with the label $y$ (waterbird or landbird). Based on the group definition, we evaluate the model via the worst test group error, i.e., $\max_g E_{(x,y) \sim g}[\ell_{0-1}(f_0(x), y)]$, where $\ell_{0-1}$ represents the 0-1 loss.

3 Learning Invariant Representations with Selective Augmentation

This section presents LISA, a simple way to improve robustness to domain shifts or subpopulation shifts. The key idea behind LISA is to encourage the model to alleviate the effects of domain-related spurious correlations by selecting data interpolation. Before detailing how to select interpolated samples, we first provide a general formulation for data interpolation.

In LISA, we perform linear interpolation between training samples. Specifically, given samples $(x_i, y_i, d_i)$ and $(x_j, y_j, d_j)$ drawn from domains $d_i$ and $d_j$, we apply mixup (Zhang et al., 2018), a simple data interpolation strategy, separately on the input features and corresponding labels as:

$$x_{\text{mix}} = \lambda x_i + (1-\lambda)x_j, \quad y_{\text{mix}} = \lambda y_i + (1-\lambda)y_j,$$

where the interpolation ratio $\lambda \in [0, 1]$ is sampled from a Beta distribution $\text{Beta}(\alpha, \beta)$. Notice that the mixup approach in equation 2 can be replaced by CutMix (Yun et al., 2019), which shows stronger empirical performance in vision-based applications. In text-based applications, we replace the feature interpolation in equation 2 with Manifold Mixup (Verma et al., 2019), where the interpolation strategy is performed on the output representation of a pre-trained model, e.g., the output of BERT (Devlin et al., 2019).

After obtaining the interpolated features and labels, we replace the original features and labels in ERM with the interpolated ones. Then, the optimization process in equation 1 is reformulated as:

$$\theta^* := \arg \min_{\theta \in \Theta} \mathbb{E}_{(x,y,d) \sim D^s}(\ell_{0-1}(f_0(x), y), y_{\text{mix}}).$$

Without additional selection strategies, vanilla mixup will regularize the model and reduce overfitting (Zhang et al., 2021b), allowing it to attain good in-distribution generalization. However, vanilla mixup may not be able to cancel out spurious correlations, causing the model to still fail at attaining good OOD generalization (see empirical comparisons in Section 4.3 and theoretical discussion in Section 5). In LISA, we instead adopt a new strategy where mixup is only applied across specific domains or groups, which leans towards learning invariant representations and thus better OOD performance. Specifically, the two kinds of sample selection strategies are presented as follows:

- **Selection Strategy I: Interpolating samples with the same label.** In selection strategy I, LISA interpolates samples with the same label but different domains (i.e., $d_i \neq d_j, y_i = y_j$). This produces datapoints that have both domains partially present, effectively eliminating spurious correlations between domain and label in cases where the pair of domains correlate differently with the label. As a result, the model should learn domain-invariant representations for each class and thus achieve better OOD robustness.

- **Selection Strategy II: Interpolating samples with the same domain.** Supposing domain identification is spuriously correlated with the label information, selection strategy II applies the interpolation strategy on samples with the same domain but different labels, i.e., $d_i = d_j, y_i \neq y_j$. Intuitively, even within the same domain, the model is supposed to generate different predicted labels since the interpolation ratio $\lambda$ is randomly sampled, corresponding to different labels $y_{\text{mix}}$. This causes the model to make predictions that are less dependent on the domain, again improving OOD robustness.
In LISA, we randomly perform selection strategies I or II during the training process with probability $p_{\text{sel}}$ and $1 - p_{\text{sel}}$ for each batch of data, where $p_{\text{sel}}$ is treated as a hyperparameter in our experiments. The choice of $p_{\text{sel}}$ depends on the number of domains and classes. Intuitively, when there are many domains but only a small number of classes, using strategy I more often (i.e., $p_{\text{sel}} > 0.5$) brings more benefits to eliminate domain effects. The pseudocode of the training procedure of LISA is shown in Algorithm 1.

Algorithm 1 Training Procedure of LISA

Require: Training data, Step size $\eta$

1: while not converge do
2: Sample $\lambda \sim \text{Beta}(\alpha, \beta)$
3: Sample a set of samples $B_1$ uniformly from the training data
4: Randomly select a sample selection strategy I or II with the probability $p_{\text{sel}}$ and $1 - p_{\text{sel}}$
5: if use selection strategy I then
6: For each sample $(x_i, y_i, d_i)$, find another one $(x_j, y_j, d_j)$ from the dataset with the same label ($y_i = y_j$) but different domains ($d_i \neq d_j$), and construct set $B_2$.
7: else if use selection strategy II then
8: For each sample $(x_i, y_i, d_i)$, find another one $(x_j, y_j, d_j)$ from the same domain ($d_i = d_j$) but different labels ($y_i \neq y_j$), constructing set $B_2$.
9: Update $\theta$ with data $\lambda B_1 + (1 - \lambda) B_2$.

4 Experiments

In this section, we conduct comprehensive experiments to evaluate the effectiveness of LISA. Specifically, we aim to answer the following questions: Q1: Compared to prior methods, can LISA improve robustness to domain shifts and subpopulation shifts (Section 4.1 and Section 4.2)? Q2: Which aspects of LISA are the most important for improving robustness (Section 4.3)? Q3: How does LISA perform with varying degrees of distribution shifts (Section 4.4)? Q4: Does LISA successfully produce invariant representations (Section 4.5)?

To answer Q1, we compare to ERM, IRM (Arjovsky et al., 2019), MMD (Li et al., 2018), DRNN (Ganin & Lempitsky, 2015), GroupDRO (Sagawa et al., 2020a), DomainMix (Xu et al., 2020), and Fish (Shi et al., 2021). Upweighting (UW) is particularly suitable for subpopulation shifts, so we also use it for comparison. We adopt the same model architectures for all approaches.

4.1 Evaluating Robustness to Domain Shifts

Experimental Setup. To study domain shifts, we study five datasets. Four datasets (Camelyon17, FMoW, RxRx1, and Amazon) are selected from WILDS (Koh et al., 2021), covering real-world distribution shifts across diverse domains (e.g., health, natural language process, and vision). Besides the WILDS data, we also apply LISA on the MetaShift datasets (Liang & Zou, 2021), constructed using the real-world images and natural heterogeneity of Visual Genome (Krishna et al., 2016). We summarize the datasets in Table 1, including domain information, evaluation metric, model architecture, and the number of classes. Detailed dataset descriptions and other training details are discussed in Appendix A.1.1 and A.1.2 respectively.

Table 1: Dataset Statistics for Domain Shifts.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Domains</th>
<th>Metric</th>
<th>Base Model</th>
<th>Num. of classes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camelyon17</td>
<td>5 hospitals</td>
<td>Avg. Acc.</td>
<td>DenseNet-121</td>
<td>2</td>
</tr>
<tr>
<td>FMoW</td>
<td>16 years x 5 regions</td>
<td>Worst-group Acc.</td>
<td>DenseNet-121</td>
<td>62</td>
</tr>
<tr>
<td>RxRx1</td>
<td>51 experimental batches</td>
<td>Avg. Acc.</td>
<td>ResNet-50</td>
<td>1,139</td>
</tr>
<tr>
<td>Amazon</td>
<td>7,676 reviewers</td>
<td>10th Percentile Acc.</td>
<td>DistilBERT-uncased</td>
<td>5</td>
</tr>
<tr>
<td>MetaShift</td>
<td>4 backgrounds</td>
<td>Worst-group Acc.</td>
<td>ResNet-50</td>
<td>2</td>
</tr>
</tbody>
</table>

Results. We report the results of domain shifts in Table 2, where the full results that include validation performance and other metrics are listed in Appendix A.1.3. According to Table 2, LISA
outperforms prior methods on all five datasets regardless of the model architecture and dataset types (i.e., image or text), demonstrating its effectiveness in improving OOD robustness by canceling out spurious correlations with augmentation. We also notice that there are no significant performance differences between ERM and other invariant representation learning methods (e.g. IRM, CORAL, DomainMix) on most datasets, which is consistent with the reported results on WILDS [Koh et al., 2021]. However, the consistent superiority of LISA over ERM still demonstrates the power of learning invariant representations in improving real-world OOD robustness. We argue that the failure of other methods in some datasets may be caused by their regularizers limiting model capacity to some extent.

During hyperparameter selection, we find that the optimal probability $p_{sel}$ of choosing the selection strategy I and II is 1.0 for these domain shifts problems, where LISA only interpolates samples with the same label. A potential reason is that the existing domain identification is weakly or even not spuriously associated with labels in most cases, which violates the assumption of selection strategy II to some extent. For example, in Camelyon17-wilds dataset [Koh et al., 2021], the presence of tumor tissue (i.e., label) mainly depends on the demographic of patients (e.g., race, gender), which shows no significant difference across hospitals (i.e., domain information). Under this scenario, performing selection strategy I to directly eliminate can potentially have more benefits.

Table 2: Main domain shifts results. LISA outperforms prior methods on all five datasets. Following the instructions of [Koh et al., 2021], we report the performance of Camelyon17 over 10 different seeds and the results of other datasets are obtained over 3 different seeds.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ERM</td>
<td>70.3 ± 6.4%</td>
<td>32.3 ± 1.25%</td>
<td>29.9 ± 0.4%</td>
<td>53.8 ± 0.8%</td>
<td>52.1 ± 0.4%</td>
</tr>
<tr>
<td>IRM</td>
<td>64.2 ± 8.1%</td>
<td>30.0 ± 1.37%</td>
<td>8.2 ± 1.1%</td>
<td>52.4 ± 0.8%</td>
<td>51.8 ± 0.8%</td>
</tr>
<tr>
<td>CORAL</td>
<td>59.5 ± 7.7%</td>
<td>31.7 ± 1.24%</td>
<td>28.4 ± 0.3%</td>
<td>52.9 ± 0.8%</td>
<td>47.9 ± 1.9%</td>
</tr>
<tr>
<td>GroupDRO</td>
<td>68.4 ± 7.3%</td>
<td>30.8 ± 0.81%</td>
<td>23.0 ± 0.3%</td>
<td>53.3 ± 0.0%</td>
<td>51.9 ± 0.7%</td>
</tr>
<tr>
<td>DomainMix</td>
<td>69.7 ± 5.5%</td>
<td>34.2 ± 0.76%</td>
<td>30.8 ± 0.4%</td>
<td>53.3 ± 0.0%</td>
<td>51.3 ± 0.5%</td>
</tr>
<tr>
<td>Fish</td>
<td>74.7 ± 7.1%</td>
<td>34.6 ± 0.18%</td>
<td>10.1 ± 1.5%</td>
<td>53.3 ± 0.0%</td>
<td>49.2 ± 2.1%</td>
</tr>
<tr>
<td><strong>LISA (ours)</strong></td>
<td><strong>77.1 ± 6.5%</strong></td>
<td><strong>35.5 ± 0.65%</strong></td>
<td><strong>31.9 ± 0.8%</strong></td>
<td><strong>54.7 ± 0.0%</strong></td>
<td><strong>54.2 ± 0.7%</strong></td>
</tr>
</tbody>
</table>

4.2 Evaluating Robustness to Subpopulation Shifts

**Evaluation Protocol.** In subpopulation shifts, we evaluate the performance on four binary classification datasets, including Colored MNIST (CMNIST), Waterbirds [Sagawa et al., 2020a], CelebA [Liu et al., 2015], and CivilComments [Borkan et al., 2019]. We summarize brief data statistics in Table 3 covering domain information, model architecture, and class information. Full dataset descriptions of subpopulation shifts are presented in Appendix A.2.1. Following Sagawa et al. (2020a), in subpopulation shifts, we use the worst-group accuracy to evaluate the performance of all approaches and the domain identifications are highly spurious correlated with the label information. For example, as suggested in Figure 1, 95% images in the Waterbirds dataset have the same background and bird type, i.e., waterbirds in water or landbirds in land. Full hyperparameter settings and training details are listed in Appendix A.2.2.

Table 3: Dataset Statistics for Subpopulation Shifts. All datasets are binary classification tasks and we use the worst group accuracy as the evaluation metric.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Domains</th>
<th>Base Model</th>
<th>Class Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMNIST</td>
<td>2 digit colors</td>
<td>ResNet-50</td>
<td>digit (0,1,2,3,4) v.s. (5,6,7,8,9)</td>
</tr>
<tr>
<td>Waterbirds</td>
<td>2 backgrounds</td>
<td>ResNet-50</td>
<td>waterbirds v.s. landbirds</td>
</tr>
<tr>
<td>CelebA</td>
<td>2 hair colors</td>
<td>ResNet-50</td>
<td>man v.s. women</td>
</tr>
<tr>
<td>CivilComments</td>
<td>8 demographic identities</td>
<td>DistilBERT-uncased</td>
<td>toxic v.s. non-toxic</td>
</tr>
</tbody>
</table>

**Results.** In Table 4 we report the overall performance of LISA and other methods. Similar to the observations in domain shifts, LISA consistently outperforms prior methods in CMNIST, CelebA,
and CivilComments. In Waterbirds, LISA outperforms other invariant representation learning methods (e.g., IRM, CORAL, DomainMix, Fish) and shows similar performance to GroupDRO. These results demonstrate the effectiveness of LISA in improving OOD robustness. In CMNIST, Waterbirds, and CelebA, we find that $p_{sel} = 0.5$ works well for choosing selection strategies I and II, while $p_{sel}$ is set as 1.0 in CivilComments. This is not surprising because it might be more beneficial to use the strategy I more often to eliminate domain effects when there are more domains, i.e., eight domains in CivilComments v.s. two domains in others.

Table 4: Results of subpopulation shifts. Here, we show the average and worst group accuracy. We repeat the experiments three times and put full results with standard deviation in Table 15.

<table>
<thead>
<tr>
<th></th>
<th>CMNIST</th>
<th>Waterbirds</th>
<th>CelebA</th>
<th>CivilComments</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Avg.</td>
<td>Worst</td>
<td>Avg.</td>
<td>Worst</td>
</tr>
<tr>
<td>ERM</td>
<td>27.8%</td>
<td>0.0%</td>
<td>97.0%</td>
<td>63.7%</td>
</tr>
<tr>
<td>UW</td>
<td>72.2%</td>
<td>66.0%</td>
<td>95.1%</td>
<td>88.0%</td>
</tr>
<tr>
<td>IRM</td>
<td>72.1%</td>
<td>70.3%</td>
<td>87.5%</td>
<td>75.6%</td>
</tr>
<tr>
<td>CORAL</td>
<td>71.8%</td>
<td>69.5%</td>
<td>90.3%</td>
<td>79.8%</td>
</tr>
<tr>
<td>GroupDRO</td>
<td>72.3%</td>
<td>68.6%</td>
<td>91.8%</td>
<td>90.6%</td>
</tr>
<tr>
<td>DomainMix</td>
<td>51.4%</td>
<td>48.0%</td>
<td>76.4%</td>
<td>53.0%</td>
</tr>
<tr>
<td>Fish</td>
<td>46.9%</td>
<td>35.6%</td>
<td>85.6%</td>
<td>64.0%</td>
</tr>
<tr>
<td>LISA (ours)</td>
<td>74.0%</td>
<td>73.3%</td>
<td>91.8%</td>
<td>89.2%</td>
</tr>
</tbody>
</table>

Table 5: Compared LISA with substitute mixup strategies in domain shifts.

<table>
<thead>
<tr>
<th></th>
<th>Camelyon17</th>
<th>FMoW</th>
<th>RxRx1</th>
<th>Amazon</th>
<th>MetaShift</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERM</td>
<td>70.3 ± 6.4%</td>
<td>32.8 ± 0.45%</td>
<td>29.9 ± 0.4%</td>
<td>53.8 ± 0.8%</td>
<td>52.1 ± 0.4%</td>
</tr>
<tr>
<td>Vanilla mixup</td>
<td>71.2 ± 5.3%</td>
<td>34.2 ± 0.45%</td>
<td>26.5 ± 0.5%</td>
<td>53.3 ± 0.0%</td>
<td>51.3 ± 0.7%</td>
</tr>
<tr>
<td>In-group mixup</td>
<td>75.5 ± 6.7%</td>
<td>32.2 ± 1.18%</td>
<td>24.4 ± 0.2%</td>
<td>53.8 ± 0.6%</td>
<td>52.7 ± 0.5%</td>
</tr>
<tr>
<td>LISA (ours)</td>
<td>77.1 ± 6.5%</td>
<td>35.5 ± 0.65%</td>
<td>31.9 ± 0.8%</td>
<td>54.7 ± 0.0%</td>
<td>54.2 ± 0.7%</td>
</tr>
</tbody>
</table>

4.3 ABLATION STUDY: IS THE PERFORMANCE GAIN FROM DATA AUGMENTATION?

In LISA, we apply selective interpolation strategies on samples either with the same label but different domains or with the same domain but different labels. Here, we explore two substitute interpolation strategies: (1) *Vanilla mixup*: In Vanilla mixup, we do not add any constraint on the sample selection, i.e., the mixup is performed on any pairs of samples; (2) *In-group mixup*: This strategy applies data interpolation on samples with the same labels and from the same domains. Notice that all the substitute interpolation strategies use the same mixup types (e.g., mixup/Manifold Mixup/CutMix) as LISA. Finally, since upweighting (UW) small groups significantly improves performance in subpopulation shifts, we also evaluate UW combined with Vanilla/In-group mixup.

The ablation results of domain shifts and subpopulation shifts are in Table 5 and Table 6 respectively. From the results, we make the following three key observations. First, compared with vanilla mixup, the performance of LISA verifies that selective data interpolation does improve the out-of-distribution robustness by canceling out the spurious correlations and encouraging invariant representation learning. The observation also corroborates our motivation that simply applying mixup does not reliably improve OOD generalization. Second, the superiority of LISA over In-group mixup verifies that only interpolating samples within each group is incapable of eliminating out the spurious information, where In-group mixup still performs the role of data augmentation. Finally, though incorporating UW significantly improves the performance of Vanilla mixup and In-group mixup in subpopulation shifts, LISA still achieves larger benefits than these enhanced substitute strategies, demonstrating its stronger power in improving OOD robustness.

4.4 EFFECT OF THE DEGREE OF DISTRIBUTION SHIFTS
Table 6: Compared LISA with substitute mixup strategies in subpopulation shifts. UW represents upweighting. Full results with standard deviation is listed in Table 16.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ERM</td>
<td>27.8%</td>
<td>97.0%</td>
<td>94.9%</td>
<td>92.2%</td>
</tr>
<tr>
<td>Vanilla mixup</td>
<td>32.6%</td>
<td>81.0%</td>
<td>95.8%</td>
<td>90.8%</td>
</tr>
<tr>
<td>Vanilla mixup + UW</td>
<td>72.2%</td>
<td>92.1%</td>
<td>91.5%</td>
<td>87.8%</td>
</tr>
<tr>
<td>In-group mixup</td>
<td>33.6%</td>
<td>88.7%</td>
<td>95.2%</td>
<td>90.8%</td>
</tr>
<tr>
<td>In-group mixup + UW</td>
<td>72.6%</td>
<td>91.4%</td>
<td>92.4%</td>
<td>84.8%</td>
</tr>
<tr>
<td>LISA (ours)</td>
<td>74.0%</td>
<td>91.8%</td>
<td>92.4%</td>
<td>89.2%</td>
</tr>
</tbody>
</table>

Table 7: Effects of the degree of distribution shifts w.r.t. the performance.

<table>
<thead>
<tr>
<th>Distance</th>
<th>0.44</th>
<th>0.71</th>
<th>1.12</th>
<th>1.43</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERM</td>
<td>80.1%</td>
<td>68.4%</td>
<td>52.1%</td>
<td>33.2%</td>
</tr>
<tr>
<td>IRM</td>
<td>79.5%</td>
<td>67.4%</td>
<td>51.8%</td>
<td>32.0%</td>
</tr>
<tr>
<td>DomainMix</td>
<td>76.0%</td>
<td>63.7%</td>
<td>51.3%</td>
<td>30.8%</td>
</tr>
<tr>
<td>GroupDRO</td>
<td>77.0%</td>
<td>68.9%</td>
<td>51.9%</td>
<td>34.2%</td>
</tr>
<tr>
<td>LISA (ours)</td>
<td>81.3%</td>
<td>69.7%</td>
<td>54.2%</td>
<td>37.5%</td>
</tr>
</tbody>
</table>

4.5 Analysis about Learned Invariance

Finally, we analyze the invariant representations learned by LISA. For each label $y$, assume the hidden representation for each domain $d$ as $H^y_d$. The invariance IV is measured by the pairwise KL divergence of distribution $P(H^y_d)$ among all domains as $IV = \frac{1}{|Y| |D|} \sum_{y \in Y} \sum_{d,d' \in D} KL(P(H^y_D | D = d) | P(H^y_D | D = d'))$, where smaller IV values indicate that the learned representations are more invariant with respect to the labels. We report the results of CMNIST and MetaShift in Table 8 where the results of IRM, DomainMix, and vanilla mixup are also reported for comparison. Our key observations are: (1) Compared with ERM, the invariance of LISA indicates its promise in improving the OOD robustness by encouraging invariant representation learning. (2) LISA has greater invariance than vanilla mixup, validating that the invariant representations are not caused by naive data interpolation. (3) LISA provides more invariant representations than regularization-based methods, i.e., IRM and DomainMix.

5 Theoretical Analysis

In this section, we provide some theoretical understandings that explain several of the empirical phenomena from the previous experiments and theoretically compare the worst-group errors of three methods: the proposed LISA, ERM, and vanilla mixup. Specifically, we consider a Gaussian mixture model with subpopulation and domain shifts, which has been widely adopted in theory to shed light upon complex machine learning phenomenon such as in [Montanari et al. (2019); Zhang et al. (2021c)]. We also note here that despite the popularity of mixup in practice, the theoretical analysis of how mixup (with or without the selection strategies) affects the misclassification error is still largely unexplored in the literature even in the simple models. As discussed in Section 2 here, we
Let $\Sigma$ classification rule is linear with slope $\Sigma$. According to the theory of Fisher’s linear discriminant analysis rule (Anderson, 1962), the optimal mixup strategy for LISA. We also denote its finite sample version by 

$$w_{st} = \max_{d \in \{B, G\}, y \in \{0, 1\}} E^{(y, d)}(b, b_0) =: \pi(d, y, i) = y, \quad \text{and the worst-group classification errors of LISA are asymptotically}$$

$$\tilde{E}_{\text{LISA}} \leq \min\{ \tilde{E}_{\text{ERM}}^{(w)}, \tilde{E}_{\text{mix}}^{(w)} \} + O_p \left( \frac{p \log n}{n} + \frac{p}{\alpha n} \right).$$

Theorem 1 implies that when $\xi$ is small (indicating that the domain has strong spurious correlation with the label) and $p = o(\alpha n)$, the worst-group classification errors of LISA are asymptotically smaller than that of ERM and vanilla mixup. In fact, our analysis shows that LISA yields a classification rule closer to the invariant classification rules by leveraging the domain information.

In the next theorem, we present the mis-classification error comparisons with domain shifts. That is, consider samples from a new unseen domain:

$$x_1^{(0, i)} = N(\mu^{(0, i)}, \Sigma), \quad x_1^{(1, i)} = N(\mu^{(1, i)}, \Sigma).$$

Let $\Delta^* = 2(\mu^{(0, *)} - \Sigma)$, where $E[x_1]$ is the mean of the training distribution, and assume $\mu^{(1, *)} - \mu^{(0, *)} = \Delta$. Let $\xi = \Delta^* / \|\Delta^*\|_2$ and $\gamma = \Delta^* / \|\Delta^*\|_2$ denote the correlation for $(\Delta^*, \Delta)$ and for $(\Delta^*, \Delta)$, respectively, with respect to $\Sigma^{-1}$. Let $E_A^{(w, st)} = \max_{y \in \{0, 1\}} E^{(y, d)}(b, b_0, A)$ and its sample version be $E_A^{(w, st)}$.
Theorem 2 (Mis-classification error comparison with domain shifts). Suppose $n$ samples are independently generated from model $(d)$, $\pi^{(B)} = \pi^{(1)} = 1/2, \pi^{(0,B)} = \pi^{(1,G)} = \alpha < 1/4$, $\max_{y,d} \|\mu_y(d)\|_2 \leq C$ and $\Sigma$ is positive definite. Suppose that $(\xi, \xi^*, \gamma)$ satisfy that $0 \leq \xi^* \leq \gamma \xi$ and $\xi \leq \min\{\gamma \|\Delta\|_{\Sigma^{-1}}, 1\} - C\alpha$ for some large enough constant $C$ and $\|\Delta\|_{\Sigma} \leq \sqrt{\max\{\text{var}(\lambda), 1/4\}}$. Then for any $p_{sel} \in [0, 1]$,
\[
\tilde{E}_{\text{LISA}}^{wst(s)} \leq \min\{\tilde{E}_{\text{ERM}}^{wst(s)}, \tilde{E}_{\text{mix}}^{wst(s)}\} + O_P\left(\frac{p \log n}{n} + \frac{p}{\alpha n}\right).
\]
Similar to Theorem 1, this result shows that when domain has strong spurious correlation with the label (corresponding to small $\xi$), such a spurious correlation leads to the downgraded performance of ERM and vanilla mixup, while our proposed LISA method is able to mitigate such an issue by selective data interpolation. Proofs of Theorem 1 and Theorem 2 are provided in Appendix B.

6 RELATED WORK AND DISCUSSION

In this paper, we focus on improving the robustness of machine learning models to domain shifts and subpopulation shifts. Here, we discuss related approaches from the following three categories:

Learning Invariant Representations with Domain Alignment. Motivated by unsupervised domain adaptation [Ben-David et al., 2010; Ganin et al., 2016], the first category of works learns invariant representations by aligning representations across domains. The major research line of this category aims to eliminate the domain dependency by minimizing the divergence of feature distributions with different distance metrics, e.g., maximum mean discrepancy [Tzeng et al., 2014; Long et al., 2015], an adversarial loss [Ganin et al., 2016; Li et al., 2018], Wasserstein distance [Zhou et al., 2020a]. Follow-up works applied data augmentation to generate more domains and enhance the consistency of representations during training [Yue et al., 2019; Zhou et al., 2020b; Xu et al., 2020; Yan et al., 2020]. Unlike these latter methods, LISA instead focuses on canceling out correlations in the dataset between the domain and the label through selective data interpolation, leading to stronger empirical performance.

Learning Invariant Representations with Invariant Predictors. Beyond using domain alignment to learning invariant representations, recent work aims to further enhance the correlations between the invariant representations and the labels [Koyama & Yamaguchi, 2020]. Representatively, motivated by casual inference, invariant risk minimization (IRM) [Arjovsky et al., 2019] aims to find a predictor that performs well across all domains. After IRM, the following works propose stronger regularizers by penalizing the variance of risks across all domains [Krueger et al., 2021], by aligning the gradient across domains [Koyama & Yamaguchi, 2020], or through game-theoretic invariant rationalization criterion [Chang et al., 2020]. Instead of using regularization, LISA eliminates spurious correlations in the data directly via data interpolation.

Group Robustness. The last category of methods combats spurious correlations and are particularly suitable for subpopulation shifts. These approaches include directly optimizing the worst-group performance with Distributionally Robust Optimization [Sagawa et al., 2020b; Zhang et al., 2021a; Zhou et al., 2021], generating samples around the minority groups [Goel et al., 2021], and balancing the majority and minority groups via reweighting [Sagawa et al., 2020b] or regularizing [Cao et al., 2019, 2020]. Here, LISA proposes a more general strategy based on data augmentation that is suitable for both domain shifts and subpopulation shifts.

7 CONCLUSION

To tackle the distribution shifts, we propose LISA, a simple and efficient algorithm, to improve the out-of-distribution robustness. LISA aims to eliminate the domain-related spurious correlations among the training set by selective sample interpolation. We evaluate the effectiveness of LISA on nine datasets under subpopulation shifts and domain shifts settings, demonstrating its promise. Besides, our detailed analysis verifies that the performance gains caused by LISA result from encouraging learning invariant representations. Our theoretical results further strengthen the superiority of LISA by showing smaller worst-group mis-classification error compared with ERM and vanilla data interpolation.
REPRODUCIBILITY STATEMENT

We conduct experiments under the setting of domain shifts and subpopulation shifts problems. In terms of the domain shifts, the results are reported in in Table 2. The dataset details are provided in Appendix A.1.1 and the training details along with the hyperparameter settings are in appendix A.1.2. Then, for the subpopulation shifts, we have included the full results including the error bounds in Table 15. The dataset details are discussed in Appendix A.2.1 while the training details and the hyperparameter settings are in Appendix A.2.2. We will release the code upon publication. Besides, The detailed proof of Theorem 1 and Theorem 2 are provided in Appendix B.

REFERENCES


**A ADDITIONAL EXPERIMENTS**

**A.1 DOMAIN SHIFTS**

**A.1.1 DATASET DETAILS**

In this section, we provide detailed descriptions of datasets used in the experiments of domain shifts.
Camelyon17 We use Camelyon17 from the WILDS benchmark [Koh et al., 2021; Bandi et al., 2018], which provides 450,000 lymph-node scans sampled from 5 hospitals. Camelyon17 is a medical image classification task where the input $x$ is a $96 \times 96$ image and the label $y$ is whether there exists tumor tissue in the image. The domain $d$ denotes the hospital that the patch was taken from. The training dataset is drawn from the first 3 hospitals, while out-of-distribution validation and out-of-distribution test datasets are sampled from the 4-th hospital and 5-th hospital respectively.

FMoW The FMoW dataset is from the WILDS benchmark [Koh et al., 2021; Christie et al., 2018] — a satellite image classification task which includes 62 classes and 80 domains (16 years x 5 regions). Concretely, the input $x$ is a $224 \times 224$ RGB satellite image, the label $y$ is one of the 62 building or land use categories, and the domain $d$ represents the year that the image was taken as well as its corresponding geographical region – Africa, the Americas, Oceania, Asia, or Europe. The train/test/validation splits are based on the time when the images are taken. Specifically, images taken before 2013 are used as the training set. Images taken between 2013 and 2015 are used as the validation set. Images taken after 2015 are used for testing.

RxRx1 RxRx1 [Koh et al., 2021; Taylor et al., 2019] from the WILDS benchmark is a cell image classification task. In the dataset, some cells have been genetically perturbed by siRNA. The goal of RxRx1 is to predict which siRNA that the cells have been treated with. Concretely, the input $x$ is an image of cells obtained by fluorescent microscopy, the label $y$ indicates which of the 1, 139 genetic treatments the cells received, and the domain $d$ denotes the experimental batches. Here, 33 different batches of images are used for training, where each batch contains one sample for each class. The out-of-distribution validation set has images from 4 experimental batches. The out-of-distribution test set has 14 experimental batches. The average accuracy on out-of-distribution test set is reported.

Amazon Each task in the Amazon benchmark [Koh et al., 2021; Ni et al., 2019] is a multi-class sentiment classification task. The input $x$ is the text of a review, the label $y$ is the corresponding star rating ranging from 1 to 5, and the domain $d$ is the corresponding reviewer. The training set has 245, 502 reviews from 1, 252 reviewers, while the out-of-distribution validation set has 100, 050 reviews from another 1, 334 reviewers. The out-of-distribution test set also has 100, 050 reviews from the rest 1, 252 reviewers. We evaluate the models by the 10th percentile of per-user accuracies in the test set.

MetaShift We use the MetaShift [Liang & Zou 2021], which is derived from Visual Genome [Krishna et al., 2016]. MetaShift leverages the natural heterogeneity of Visual Genome to provide many distinct data distributions for a given class (e.g. “cats with cars” or “cats in bathroom” for the “cat” class). A key feature of MetaShift is that it provides explicit explanations of the dataset correlation and a distance score to measure the degree of distribution shift between any pair of sets. We adopt the “Cat vs. Dog” task in MetaShift, where we evaluate the model on the “dog(shelf)” domain with 306 images, and the “cat(shelf)” domain with 235 images. The training data for the “Cat” class is the cat(sofa + bed), including cat(sofa) domain and cat(bed) domain. MetaShift provides 4 different sets of training data for the “Dog” class in an increasingly challenging order, i.e., increasing the amount of distribution shift. Specifically, dog(cabinet + bed), dog(bag + box), dog(bench + bike), dog(boat + surfboard) are selected for training, where their corresponding distances to dog(shelf) are 0.44, 0.71, 1.12, 1.43.

A.1.2 Training Details

Follow WILDS [Koh et al., 2021], we adopt pre-trained DenseNet121 [Huang et al., 2017] for Camelyon17 and FMoW datasets, ResNet-50 [He et al., 2016] for RxRx1 and MetaShift datasets, and DistilBert [Sanh et al., 2019] for Amazon datasets. In domain shifts, the selection probability $p_{sel}$ is set as 1.0, where we only use selection strategy I and apply data interpolation on samples with the same label. Besides, in selection strategy I, we use a more general setting in the experiments of domain shifts, where samples with the same label are interpolated regardless of their domain information. This more general strategy yields the best empirical performance. As discussed in the ”Results” paragraph in Section 4.1, the existing domain identifications are potentially weakly or even not spuriously correlated with the label information, and therefore using domain identification in interpolation does not benefit us that much. Instead, in such scenarios, enlarging the interpolation
scope might bring more benefits. In each training iteration, we first draw a batch of samples $B_1$ from the training set. With $B_1$, we then select another sample batch $B_2$ with same labels as $B_1$ for data interpolation. The interpolation ratio $\lambda$ is drawn from the distribution Beta(2, 2). We use the same image transformers as Koh et al. (2021), and all other hyperparameter settings are listed in Table 9.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Camelyon17</th>
<th>FMoW</th>
<th>RxRx1</th>
<th>Amazon</th>
<th>MetaShift</th>
</tr>
</thead>
<tbody>
<tr>
<td>Learning rate</td>
<td>1e-4</td>
<td>1e-4</td>
<td>1e-3</td>
<td>2e-6</td>
<td>1e-3</td>
</tr>
<tr>
<td>Weight decay</td>
<td>0</td>
<td>0</td>
<td>1e-5</td>
<td>0</td>
<td>1e-4</td>
</tr>
<tr>
<td>Scheduler</td>
<td>n/a</td>
<td>n/a</td>
<td>Cosine Warmup</td>
<td>n/a</td>
<td>n/a</td>
</tr>
<tr>
<td>Batch size</td>
<td>32</td>
<td>32</td>
<td>72</td>
<td>8</td>
<td>16</td>
</tr>
<tr>
<td>Type of mixup</td>
<td>CutMix</td>
<td>CutMix</td>
<td>CutMix</td>
<td>ManifoldMix</td>
<td>CutMix</td>
</tr>
<tr>
<td>Architecture</td>
<td>DenseNet121</td>
<td>DenseNet121</td>
<td>ResNet50</td>
<td>DistilBert</td>
<td>ResNet50</td>
</tr>
<tr>
<td>Optimizer</td>
<td>SGD</td>
<td>Adam</td>
<td>Adam</td>
<td>Adam</td>
<td>SGD</td>
</tr>
<tr>
<td>Maximum Epoch</td>
<td>2</td>
<td>5</td>
<td>90</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>Strategy sel. prob. $p_{sel}$</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
</tbody>
</table>

A.1.3 FULL RESULTS OF WILDS DATA

Follow Koh et al. (2021), we reported more results on WILDS datasets in Table 10 - Table 13, including validation performance and the results of other metrics. According to these additional results, we could see that LISA outperforms other baseline approaches in all scenarios. Particularly, we here discuss two additional findings: (1) In Camelyon dataset, the test data is much more visually distinctive compared with the validation data, resulting in the large gap ($\sim 10\%$) between validation and test performance of ERM (see Table 10). However, LISA significantly reduces the performance gap between the validation and test sets, showing its promise in improving OOD robustness; (2) In Amazon dataset, though LISA performs worse than ERM in average accuracy, it achieves the best accuracy at the 10th percentile, which is regarded as a more common and important metric to evaluate whether models perform consistently well across all users (Koh et al., 2021).

Table 10: Full Results of Camelyon17. We report both validation accuracy and test accuracy.

<table>
<thead>
<tr>
<th></th>
<th>Validation Acc.</th>
<th>Test Acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERM</td>
<td>84.9 ± 3.1%</td>
<td>70.3 ± 6.4%</td>
</tr>
<tr>
<td>IRM</td>
<td><strong>86.2 ± 1.4%</strong></td>
<td>64.2 ± 8.1%</td>
</tr>
<tr>
<td>Coral</td>
<td><strong>86.2 ± 1.4%</strong></td>
<td>59.5 ± 7.7%</td>
</tr>
<tr>
<td>GroupDRO</td>
<td>85.5 ± 2.2%</td>
<td>68.4 ± 7.3%</td>
</tr>
<tr>
<td>DomainMix</td>
<td>83.5 ± 1.1%</td>
<td>69.7 ± 5.5%</td>
</tr>
<tr>
<td>Fish</td>
<td>83.9 ± 1.2%</td>
<td>74.7 ± 7.1%</td>
</tr>
<tr>
<td><strong>LISA (ours)</strong></td>
<td>81.8 ± 1.3%</td>
<td><strong>77.1 ± 6.5%</strong></td>
</tr>
</tbody>
</table>

Table 11: Full Results of FMoW. Here, we report the average accuracy and the worst-domain accuracy on both validation and test sets.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ERM</td>
<td><strong>59.5 ± 0.37%</strong></td>
<td>48.9 ± 0.62%</td>
<td>53.0 ± 0.55%</td>
<td>32.3 ± 1.25%</td>
</tr>
<tr>
<td>IRM</td>
<td>57.4 ± 0.37%</td>
<td>47.5 ± 1.57%</td>
<td>50.8 ± 0.13%</td>
<td>30.0 ± 1.37%</td>
</tr>
<tr>
<td>Coral</td>
<td>56.9 ± 0.25%</td>
<td>47.1 ± 0.43%</td>
<td>50.5 ± 0.36%</td>
<td>31.7 ± 1.24%</td>
</tr>
<tr>
<td>GroupDRO</td>
<td>58.8 ± 0.19%</td>
<td>46.5 ± 0.25%</td>
<td>52.1 ± 0.50%</td>
<td>30.8 ± 0.81%</td>
</tr>
<tr>
<td>DomainMix</td>
<td>58.6 ± 0.29%</td>
<td>48.9 ± 1.15%</td>
<td>51.6 ± 0.19%</td>
<td>34.2 ± 0.76%</td>
</tr>
<tr>
<td>Fish</td>
<td>57.8 ± 0.15%</td>
<td><strong>49.5 ± 2.34%</strong></td>
<td>51.8 ± 0.32%</td>
<td>34.6 ± 0.18%</td>
</tr>
<tr>
<td><strong>LISA (ours)</strong></td>
<td>58.7 ± 0.92%</td>
<td>48.7 ± 0.74%</td>
<td><strong>52.8 ± 0.94%</strong></td>
<td><strong>35.5 ± 0.65%</strong></td>
</tr>
</tbody>
</table>

14
Table 12: Full Results of RxRx1. ID: in-distribution; OOD: out-of-distribution

<table>
<thead>
<tr>
<th></th>
<th>Validation Acc.</th>
<th>Test ID Acc.</th>
<th>Test OOD Acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERM</td>
<td>19.4 ± 0.2%</td>
<td>35.9 ± 0.4%</td>
<td>29.9 ± 0.4%</td>
</tr>
<tr>
<td>IRM</td>
<td>5.6 ± 0.4%</td>
<td>9.9 ± 1.4%</td>
<td>8.2 ± 1.1%</td>
</tr>
<tr>
<td>Coral</td>
<td>18.5 ± 0.4%</td>
<td>34.0 ± 0.3%</td>
<td>28.4 ± 0.3%</td>
</tr>
<tr>
<td>GroupDRO</td>
<td>15.2 ± 0.1%</td>
<td>28.1 ± 0.3%</td>
<td>23.0 ± 0.3%</td>
</tr>
<tr>
<td>DomainMix</td>
<td>19.3 ± 0.7%</td>
<td>39.8 ± 0.2%</td>
<td>30.8 ± 0.4%</td>
</tr>
<tr>
<td>Fish</td>
<td>7.5 ± 0.6%</td>
<td>12.7 ± 1.9%</td>
<td>10.1 ± 1.5%</td>
</tr>
<tr>
<td>LISA (ours)</td>
<td>20.1 ± 0.4%</td>
<td>41.2 ± 1.0%</td>
<td>31.9 ± 0.8%</td>
</tr>
</tbody>
</table>

Table 13: Full Results of Amazon. Both the average accuracy and the 10th Percentile accuracy are reported.

<table>
<thead>
<tr>
<th></th>
<th>Validation</th>
<th>10-th Per.</th>
<th>Test</th>
<th>10-th Per.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERM</td>
<td>72.7 ± 0.1%</td>
<td>55.2 ± 0.7%</td>
<td>71.9 ± 0.1%</td>
<td>53.8 ± 0.8%</td>
</tr>
<tr>
<td>IRM</td>
<td>71.5 ± 0.3%</td>
<td>54.2 ± 0.8%</td>
<td>70.5 ± 0.3%</td>
<td>52.4 ± 0.8%</td>
</tr>
<tr>
<td>Coral</td>
<td>72.0 ± 0.3%</td>
<td>54.7 ± 0.0%</td>
<td>70.0 ± 0.6%</td>
<td>52.9 ± 0.8%</td>
</tr>
<tr>
<td>GroupDRO</td>
<td>70.7 ± 0.6%</td>
<td>54.7 ± 0.0%</td>
<td>70.0 ± 0.6%</td>
<td>53.3 ± 0.0%</td>
</tr>
<tr>
<td>DomainMix</td>
<td>71.9 ± 0.2%</td>
<td>54.7 ± 0.0%</td>
<td>71.1 ± 0.1%</td>
<td>53.3 ± 0.0%</td>
</tr>
<tr>
<td>Fish</td>
<td>72.5 ± 0.0%</td>
<td>54.7 ± 0.0%</td>
<td>71.7 ± 0.1%</td>
<td>53.3 ± 0.0%</td>
</tr>
<tr>
<td>LISA (ours)</td>
<td>71.2 ± 0.3%</td>
<td>55.1 ± 0.61%</td>
<td>70.6 ± 0.3%</td>
<td>54.7 ± 0.0%</td>
</tr>
</tbody>
</table>

A.2 Subpopulation Shifts

A.2.1 Dataset Details

We detail the data descriptions of subpopulation shifts in the follows:

Colored MNIST (CMNIST): We classify MNIST digits from 2 classes, where classes 0 and 1 indicate original digits (0,1,2,3,4) and (5,6,7,8,9). The color is treated as a spurious attribute. Concretely, in the training set, the proportion between red samples and green samples is 8:2 in class 0, while the proportion is set as 2:8 in class 1. In the validation set, the proportion between green and red samples is 1:1 for all classes. In the test set, the proportion between green and red samples is 1:9 in class 0, while the ratio is 9:1 in class 1. The data sizes of train, validation, and test sets are 30000, 10000, and 20000, respectively.

Waterbirds (Sagawa et al., 2020a): The Waterbirds dataset aims to classify birds as “waterbird” or “landbird”, where each bird image is spuriously associated with the background “water” or “land”. Waterbirds is a synthetic dataset where each image is composed by pasting a bird image sampled from CUB dataset (Wah et al., 2011) to a background drawn from the Places dataset (Zhou et al., 2017). The bird categories in CUB are stratified as land birds or water birds. Specifically, the following bird species are selected to construct the waterbird class: albatross, auklet, cormorant, frigatebird, fulmar, gull, jaeger, kittiwake, pelican, puffin, tern, gadwall, grebe, mallard, merganser, guillemot, or Pacific loon. All other bird species are combined as the landbird class. We define (land background, waterbird) and (water background, landbird) are minority groups. There are 4,795 training samples while only 56 samples are “waterbirds on land” and 184 samples are “landbirds on water”. The remaining training data include 3,498 samples from “landbirds on land”, and 1,057 samples from “waterbirds on water”.

CelebA (Liu et al., 2015; Sagawa et al., 2020a): For the CelebA data (Liu et al., 2015), we follow the data preprocess procedure from Sagawa et al. (2020a). CelebA defines an image classification task where the input is a face image of celebrities and the classification label is its corresponding hair color – “blond” or “not blond.” The label is spuriously correlated with gender, i.e., male or female. In CelebA, the minority groups are (blond, male) and (not blond, female). The number of samples for each group are 71,629 “dark hair, female”, 66,874 “dark hair, male”, 22,880 “blond hair, female”, 1,387 “blond hair, male”.

15
CivilComments (Borkan et al., 2019; Koh et al., 2021): We use CivilComments from the WILDS benchmark (Koh et al., 2021). CivilComments is a text classification task, aiming to predict whether an online comment is toxic or non-toxic. The spurious domain identifications are defined as the demographic features, including male, female, LGBTQ, Christian, Muslim, other religion, Black, and White. CivilComments contains 450,000 comments collected from online articles. The number of samples for training, validation, and test are 269,038, 45,180, and 133,782, respectively. The readers may kindly refer to Table 17 in Koh et al. (2021) for the detailed group information.

A.2.2 Training Details

We adopt pre-trained ResNet-50 (He et al., 2016) and BERT (Sanh et al., 2019) as the model for image data (i.e., CMNIST, Waterbirds, CelebA) and text data (i.e., CivilComments), respectively. In each training iteration, we sample a batch of data per group. For sample selection strategy I, we randomly apply mixup on sample batches with the same labels but different domains. For sample selection strategy II, we instead apply mixup on sample batches with the same domain but different labels. The interpolation ratio $\lambda$ is sampled from the distribution $\text{Beta}(2, 2)$. All hyperparameters are listed in Table 14.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>CMNIST</th>
<th>Waterbirds</th>
<th>CelebA</th>
<th>CivilComments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Learning rate</td>
<td>1e-3</td>
<td>1e-3</td>
<td>1e-4</td>
<td>1e-5</td>
</tr>
<tr>
<td>Weight decay</td>
<td>1e-4</td>
<td>1e-4</td>
<td>1e-4</td>
<td>0</td>
</tr>
<tr>
<td>Scheduler</td>
<td>n/a</td>
<td>n/a</td>
<td>n/a</td>
<td>n/a</td>
</tr>
<tr>
<td>Batch size</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>8</td>
</tr>
<tr>
<td>Type of mixup</td>
<td>mixup</td>
<td>mixup</td>
<td>CutMix</td>
<td>ManifoldMix</td>
</tr>
<tr>
<td>Architecture</td>
<td>ResNet50</td>
<td>ResNet50</td>
<td>ResNet50</td>
<td>DistilBert</td>
</tr>
<tr>
<td>Optimizer</td>
<td>SGD</td>
<td>SGD</td>
<td>SGD</td>
<td>Adam</td>
</tr>
<tr>
<td>Maximum Epoch</td>
<td>300</td>
<td>300</td>
<td>50</td>
<td>3</td>
</tr>
<tr>
<td>Strategy sel. prob. $p_{sel}$</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>1.0</td>
</tr>
</tbody>
</table>

A.2.3 Additional Results

In this section, we have added the full results of subpopulation shifts in Table 15 and Table 16.

B Proofs of Theorem 1 and Theorem 2

Outline of the proof. We will first find the mis-classification errors based on the population version of OLS with different mixup strategies. Next, we will develop the convergence rate of the empirical OLS based on $n$ samples towards its population version. These two steps together give us the empirical mis-classification errors of different methods. We will separately show that the upper bounds in Theorem 1 and Theorem 2 hold for two strategies of LISA and hence hold for any $p_{sel} \in [0, 1]$. Let L1 denote selection strategy I of LISA method and L2 denote selection strategy II of LISA method.

Let $\pi_1 = \mathbb{P}(y_i = 1)$ and $\pi_0 = \mathbb{P}(y_i = 0)$ denote the marginal class proportions in the training samples. Let $\pi_B = \mathbb{P}(d_i = B)$ and $\pi_G = \mathbb{P}(d_i = G)$ denote the marginal subpopulation proportions in the training samples. Let $\pi_{G|1} = \mathbb{P}(d_i = G|y_i = 1)$ and define $\pi_{G|0}$, $\pi_{B|1}$, and $\pi_{B|0}$ similarly.

We consider the setting where $\alpha := \pi^{(1,G)} = \pi^{(0,B)}$ is relatively small and $\pi^{(1)} = \pi^{(0)} = \pi^{(G)} = \pi^{(B)} = 1/2$.

B.1 Decomposing the Loss Function

Recall that $\Delta = \mu^{(1,G)} - \mu^{(0,G)} = \mu^{(1,B)} - \mu^{(0,B)}$. We further define $\tilde{\Delta} = \mu^{(1)} - \mu^{(0)}$, $\theta^{(G)} = \mu^{(0,G)} - \mathbb{E}[x]$, and $\theta^{(B)} = \mu^{(0,B)} - \mathbb{E}[x]$. For the mixup estimators, we will repeatedly use the fact that $\lambda_i$ has a symmetric distribution with support $[0, 1]$. 


Table 15: Full results of subpopulation shifts with standard deviation. All the results are performed with three random seed.

<table>
<thead>
<tr>
<th></th>
<th>CMNIST</th>
<th>Waterbirds</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Avg.</td>
<td>Worst</td>
</tr>
<tr>
<td>ERM</td>
<td>27.8 ± 1.9%</td>
<td>0.0 ± 0.0%</td>
</tr>
<tr>
<td>UW</td>
<td>72.2 ± 1.1%</td>
<td>66.0 ± 0.7%</td>
</tr>
<tr>
<td>IRM</td>
<td>72.1 ± 1.2%</td>
<td>70.3 ± 0.8%</td>
</tr>
<tr>
<td>Coral</td>
<td>71.8 ± 1.7%</td>
<td>69.5 ± 0.9%</td>
</tr>
<tr>
<td>GroupDRO</td>
<td>72.3 ± 1.2%</td>
<td>68.6 ± 0.8%</td>
</tr>
<tr>
<td>DomainMix</td>
<td>51.4 ± 1.3%</td>
<td>48.0 ± 1.3%</td>
</tr>
<tr>
<td>Fish</td>
<td>46.9 ± 1.4%</td>
<td>35.6 ± 1.7%</td>
</tr>
<tr>
<td><strong>LISA (ours)</strong></td>
<td>74.0 ± 0.1%</td>
<td><strong>73.3 ± 0.2%</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>CelebA</th>
<th>CivilComments</th>
<th>Avg.</th>
<th>Worst</th>
<th>Avg.</th>
<th>Worst</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERM</td>
<td>94.9 ± 0.2%</td>
<td>47.8 ± 3.7%</td>
<td>92.2 ± 0.1%</td>
<td>56.0 ± 3.6%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UW</td>
<td>92.9 ± 0.2%</td>
<td>83.3 ± 2.8%</td>
<td>89.8 ± 0.5%</td>
<td>69.2 ± 0.9%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IRM</td>
<td>94.0 ± 0.4%</td>
<td>77.8 ± 3.9%</td>
<td>88.8 ± 0.7%</td>
<td>66.3 ± 2.1%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Coral</td>
<td>93.8 ± 0.3%</td>
<td>76.9 ± 3.6%</td>
<td>88.7 ± 0.5%</td>
<td>65.6 ± 1.3%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GroupDRO</td>
<td>92.1 ± 0.4%</td>
<td>87.2 ± 1.6%</td>
<td>89.9 ± 0.5%</td>
<td>70.0 ± 2.0%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DomainMix</td>
<td>93.4 ± 0.1%</td>
<td>65.6 ± 1.7%</td>
<td>90.9 ± 0.4%</td>
<td>63.6 ± 2.5%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fish</td>
<td>93.1 ± 0.3%</td>
<td>61.2 ± 2.5%</td>
<td>89.8 ± 0.4%</td>
<td>71.1 ± 0.4%</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>LISA (ours)</strong></td>
<td>92.4 ± 0.4%</td>
<td><strong>89.3 ± 1.1%</strong></td>
<td>89.2 ± 0.9%</td>
<td><strong>72.6 ± 0.1%</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 16: Full table of the comparison between LISA and other substitute mixup strategies in subpopulation shifts. UW represents upweighting.

<table>
<thead>
<tr>
<th></th>
<th>CMNIST</th>
<th>Waterbirds</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Avg.</td>
<td>Worst</td>
</tr>
<tr>
<td>ERM</td>
<td>27.8 ± 1.9%</td>
<td>0.0 ± 0.0%</td>
</tr>
<tr>
<td>Vanilla mixup</td>
<td>32.6 ± 3.1%</td>
<td>3.1 ± 2.4%</td>
</tr>
<tr>
<td>Vanilla mixup + UW</td>
<td>72.2 ± 0.7%</td>
<td>71.8 ± 0.1%</td>
</tr>
<tr>
<td>In-group Group</td>
<td>33.6 ± 1.9%</td>
<td>24.0 ± 1.1%</td>
</tr>
<tr>
<td>In-group + UW</td>
<td>72.6 ± 0.1%</td>
<td>71.6 ± 0.2%</td>
</tr>
<tr>
<td><strong>LISA (ours)</strong></td>
<td>74.0 ± 0.1%</td>
<td><strong>73.3 ± 0.2%</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>CelebA</th>
<th>CivilComments</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Avg.</td>
<td>Worst</td>
</tr>
<tr>
<td>ERM</td>
<td>94.9 ± 0.2%</td>
<td>47.8 ± 3.7%</td>
</tr>
<tr>
<td>Vanilla mixup</td>
<td>95.8 ± 0.0%</td>
<td>46.4 ± 0.5%</td>
</tr>
<tr>
<td>Vanilla mixup + UW</td>
<td>91.5 ± 0.2%</td>
<td>88.0 ± 0.3%</td>
</tr>
<tr>
<td>Within Group</td>
<td>95.2 ± 0.3%</td>
<td>58.3 ± 0.9%</td>
</tr>
<tr>
<td>Within Group + UW</td>
<td>92.4 ± 0.4%</td>
<td>87.8 ± 0.6%</td>
</tr>
<tr>
<td><strong>LISA (ours)</strong></td>
<td>92.4 ± 0.4%</td>
<td><strong>89.3 ± 1.1%</strong></td>
</tr>
</tbody>
</table>

For ERM estimator based on \((X, y)\), where \(b_0 = \frac{1}{2} - \mathbb{E}[x_i]Tb\), we have

\[
(\mu^{(0,G)})Tb + b_0 = (\mu^{(0,G)} - \mathbb{E}[x_i])Tb + \frac{1}{2} \\
= (\theta^{(G)})Tb + \mathbb{E}[y_i] \\
(\mu^{(1,G)})Tb + b_0 = (\mu^{(1,G)} - \mathbb{E}[x_i])Tb + \frac{1}{2} \\
= \Delta Tb + (\theta^{(G)})Tb + \mathbb{E}[y_i],
\]

For ERM estimator based on \((X, y)\), where \(b_0 = \frac{1}{2} - \mathbb{E}[x_i]Tb\), we have

\[
(\mu^{(0,G)})Tb + b_0 = (\mu^{(0,G)} - \mathbb{E}[x_i])Tb + \frac{1}{2} \\
= (\theta^{(G)})Tb + \mathbb{E}[y_i] \\
(\mu^{(1,G)})Tb + b_0 = (\mu^{(1,G)} - \mathbb{E}[x_i])Tb + \frac{1}{2} \\
= \Delta Tb + (\theta^{(G)})Tb + \mathbb{E}[y_i],
\]
Notice that based on the estimator $b, b_0$

$$E^{(1,d)}(b, b_0) = \Phi\left(\frac{-\Delta^T b - (\theta^{(d)})^T b}{\sqrt{b^T \Sigma b}}\right) \text{ and } E^{(0,d)}(b, b_0) = \Phi\left(\frac{\theta^{(d)}^T b}{\sqrt{b^T \Sigma b}}\right).$$

**B.2 Classification Errors of Four Methods with Infinite Training Samples**

We first provide the limit of the classification errors when $n \to \infty$.

**B.2.1 Baseline Method: ERM**

For the training data, it is easy to show that

$$var(x) = \mathbb{E}[var(x|y)] + var(\mathbb{E}[x|y])$$

$$\sigma + \mathbb{E}[var(\mathbb{E}[x|y,D]|y)] + var((\mu^{(1)} - \mu^{(0)})y)$$

$$\sigma + \mathbb{E}[var(\mu^{(0,B)} - \mu^{(0,G)}) \mathbb{1}(D = B)|y] + \Delta \sigma^2 \pi^{(1)} \pi^{(0)}$$

$$= \sigma^2 \mu B \pi^{(1)} \pi^{(0)}.$$

For $a_0 = \frac{1}{2}(\pi_{B1}\pi_{G1} + \pi_{B0}\pi_{G0})$ and $\Delta_0 = \mu^{(0,G)} - \mu^{(0,B)}$, the ERM has slope and intercept being

$$b = var(x)^{-1}cov(x, y)$$

$$= (\sigma + a_0 \Delta^{(2)})^{-1} \Delta$$

$$= \sigma^{-1} \Delta - \sigma^{-1} \Delta_0 \cdot \frac{a_0 \Delta \Sigma^{-1} \Delta_0}{1 + a_0 \Delta \Sigma^{-1} \Delta_0}$$

$$b_0 = \mathbb{E}[y] - \mathbb{E}[x^T b].$$

In the extreme case where $\pi_{0,B} = \pi_{1,G} = 0$, we have

$$\Delta = \mu^{(1,B)} - \mu^{(0,G)}, \theta^{(G)} = -\frac{1}{2} \Delta, \theta^{(B)} = \frac{1}{2} \Delta - \Delta_0, \text{ and } \Delta_0 = \Delta - \Delta.$$

Hence,

$$E_{0}^{(wst)} = \max \left\{ \Phi\left(\frac{(\frac{1}{2} \Delta - \Delta)^T b}{\sqrt{b^T \Sigma b}}\right), \Phi\left(\frac{-\frac{1}{2} \Delta^T b}{\sqrt{b^T \Sigma b}}\right) \right\},$$

where $b$ is computed via ERM.

**B.2.2 Baseline Method: Vanilla Mixup**

The vanilla mixup does not use the group information. Let $i_1$ be a random draw from $\{1, \ldots, n\}$. Let $i_2$ be a random draw from $\{1, \ldots, n\}$ independent of $i_1$. Let

$$\tilde{y}_i = \lambda_i y_{i_1} + (1 - \lambda_i) y_{i_2}$$

and

$$\tilde{x}_i = \lambda_i x_{i_1} + (1 - \lambda_i) x_{i_2}.$$

We can find that

$$cov(\tilde{x}_i, \tilde{y}_i) = cov(\lambda_i x_{i_1} + (1 - \lambda_i) x_{i_2}, \lambda_i y_{i_1} + (1 - \lambda_i) y_{i_2})$$

$$= cov(\lambda_i x_{i_1}, \lambda_i y_{i_1}) + cov((1 - \lambda_i) x_{i_2}, (1 - \lambda_i) y_{i_2})$$

$$= (\mathbb{E}[\lambda_i^2] + \mathbb{E}[(1 - \lambda_i)^2]) cov(x_i, y_i).$$

$$cov(\tilde{x}_i) = (\mathbb{E}[\lambda_i^2] + \mathbb{E}[(1 - \lambda_i)^2]) cov(x_i).$$
Where \( d \) is a random draw from \( \{ l : y_l = y_i, D_l = G \} \) and \( i_2 \) is a random draw from \( \{ l : y_l = y_i, D_l = B \} \). Then we perform OLS based on \((x_i^{(\lambda)}, y_i), i = 1, \ldots, n\).

We can calculate that

\[
\text{cov}(x_i^{(\lambda)}, y_i) = \text{cov}(\mathbb{E}[x_i^{(\lambda)}|y_i], y_i) = \text{cov}(\frac{1}{2} \mu^{(G)}, \frac{1}{2} \mu^{(B)}, y_i)
\]

\[
\text{var}(x_i^{(\lambda)}) = \mathbb{E}[\text{cov}(x_i^{(\lambda)}, y_i)] + \text{cov}(\mathbb{E}[x_i^{(\lambda)}|y_i], \lambda_i)
\]

\[
= 2\mathbb{E}[\lambda_i^2] \Sigma + \text{cov}(\lambda_i(\mu^{(G)} - \mu^{(B)}), \Delta y_i)
\]

\[
= 2\mathbb{E}[\lambda_i^2] \Sigma + \text{var}(\lambda_i)(\mu^{(G)} - \mu^{(B)})^2 + \pi(1) \pi(0) \Delta^2.
\]

Notice that

\[
\mathbb{E}[x_i^{(\lambda)}] = \frac{1}{4} \mu^{(G)} + \mu^{(B)} + \mu^{(B)} + \mu^{(G)} = \frac{1}{2} (\mu^{(G)} + \mu^{(B)}) = \mathbb{E}[x_i].
\]

For \( a_{L1} = \text{var}(\lambda_i)/(2\mathbb{E}[\lambda_i^2]) \) and \( \Delta_0 = \mu^{(G)} - \mu^{(B)} \), the OLS has slope and intercept being

\[
b = \text{var}(x_i^{(\lambda)})^{-1} \text{cov}(x_i^{(\lambda)}, y)
\]

\[
\propto (\Sigma + \frac{\text{var}(\lambda_i)}{\mathbb{E}[\lambda_i^2]}(\mu^{(G)} - \mu^{(B)})^2)^{-1} \Delta
\]

\[
\propto \Sigma^{-1} \Delta - \Sigma^{-1} \Delta_0 \cdot \frac{a_{L1} \Delta^T \Sigma^{-1} \Delta_0}{1 + a_{L1} \Delta^T \Sigma^{-1} \Delta_0}
\]

\[
b_0 = \mathbb{E}[y] - \mathbb{E}[(x_i^{(\lambda)})^T b].
\]

\[
E^{(\text{wst})}_{L1} = \max \{ \Phi \left( \frac{\frac{1}{2} \Delta - \Delta)^T b}{\sqrt{b^T \Sigma b}} \right), \Phi \left( \frac{-\frac{1}{2} \Delta^T b}{\sqrt{b^T \Sigma b}} \right) \},
\]

where \( b \) is computed based on \((x_i^{(\lambda)}, y_i), i = 1, \ldots, n\).

### B.4 LISA with Selection Strategy II (LISA-II): mixup within each domain

The interpolated sample can be written as

\[
(\tilde{y}_i, \tilde{x}_i) = (\lambda_i, \lambda_i x_i^{(1,G)} + (1 - \lambda_i) x_i^{(0,G)}) \quad \text{if} \quad d_i = G
\]

\[
(\tilde{y}_i, \tilde{x}_i) = (\lambda_i, \lambda_i x_i^{(1,B)} + (1 - \lambda_i) x_i^{(0,B)}) \quad \text{if} \quad d_i = B.
\]

where \( i_1 \) is a random draw from \( \{ l : d_l = d_i, y_i = 1 \} \) and \( i_2 \) is a random draw from \( \{ l : d_l = d_i, y_i = 0 \} \).

We consider regress \( \tilde{y}_i \) on \( \tilde{x}_i \).

\[
\text{cov}(\tilde{x}_i, \tilde{y}_i | d_i = G) = \text{cov}(\mathbb{E}[\tilde{x}_i | \tilde{y}_i, d_i = G], \tilde{y}_i | d_i = G) = \text{cov}(\tilde{y}_i)(\mu^{(1,G)} - \mu^{(0,G)})
\]

\[
\text{var}(\tilde{x}_i | d_i = G) = \mathbb{E} [\text{var}(\tilde{x}_i | \lambda_i, D_i = G) | d_i = G] + \text{var}(\mathbb{E} [\tilde{x}_i | \lambda_i, D_i = G] | D_i = G)
\]

\[
= 2\mathbb{E}[\lambda_i^2] \Sigma + \text{var}(\lambda_i \mu^{(1,G)} + (1 - \lambda_i) \mu^{(0,G)}) | d_i = G
\]

\[
= 2\mathbb{E}[\lambda_i^2] \Sigma + \text{var}(\tilde{y}_i) \Delta^2.
\]
We further have

\[
\text{cov}(\hat{x}_i, \hat{y}_i) = \mathbb{E}[\text{cov}(\tilde{x}_i, \tilde{y}_i|d_i)] + \text{cov}(\mathbb{E}[\tilde{x}_i|d_i], \mathbb{E}[\tilde{y}_i|d_i])
\]
\[
= \text{cov}(\tilde{x}_i^{(G)}, \tilde{y}_i^{(G)})\pi^{(G)} + \text{cov}(\tilde{x}_i^{(B)}, \tilde{y}_i^{(B)})\pi^{(B)}
\]
\[
= \text{var}(\tilde{y}_i)(\mu^{(1,G)} - \mu^{(0,G)})\pi^{(G)} + \text{var}(\tilde{y}_i)(\mu^{(1,B)} - \mu^{(0,B)})\pi^{(B)}
\]
\[
= \text{var}(\tilde{y}_i)\Delta.
\]

Moreover,

\[
\text{var}(\hat{x}_i) = \mathbb{E}[\text{var}(\tilde{x}_i|d_i)] + \text{var}(\mathbb{E}[\tilde{x}_i|d_i])
\]
\[
= \text{var}(\tilde{x}_i^{(G)})\pi^{(G)} + \text{var}(\tilde{x}_i^{(B)})\pi^{(B)} + (\mathbb{E}[\tilde{x}_i^{(G)}] - \mathbb{E}[\tilde{x}_i^{(B)}])\otimes \pi^{(G)}\pi^{(B)}
\]
\[
= 2\mathbb{E}[\lambda_i^2]\Sigma + \text{var}(\lambda_i)\Delta \otimes \pi^{(G)}(\mu^{(0,G)} - \mu^{(0,B)})\otimes \pi^{(B)}.
\]

Slope:

\[
b = \text{var}(\tilde{x}_i)^{-1}\text{cov}(\tilde{x}_i, \tilde{y}_i)
\]
\[
\propto (\Sigma + a_{L2}\Delta_0 \otimes \Delta_0)^{-1}\Delta
\]
\[
= \Sigma^{-1}\Delta - \Sigma^{-1}\Delta_0 \cdot \frac{a_{L2}(\Delta_0^T\Sigma^{-1}\Delta)}{1 + a_{L2}(\Delta_0^T\Sigma^{-1}\Delta_0)},
\]

where \(a_{L2} = \frac{\pi^{(B)}\pi^{(G)}}{2\mathbb{E}[\lambda_i^2]}\).

Moreover, \(b_0 = \mathbb{E}[\tilde{y}_i] - \mathbb{E}[\tilde{x}_i]^T b = \frac{1}{2} - \mathbb{E}[\tilde{x}_i]^T b\). Notice that

\[
\mathbb{E}[\tilde{x}_i] = \frac{1}{4}(\mu^{(0,G)} + \mu^{(1,G)} + \mu^{(0,B)} + \mu^{(1,B)})
\]
\[
= \frac{1}{4}(2\mu^{(0,G)} + \Delta + 2\mu^{(1,B)} - \Delta)
\]
\[
= \frac{1}{2}(\mu^{(0,G)} + \mu^{(1,B)}) = \mathbb{E}[x_i].
\]

Hence,

\[
E_{L2}^{(wst)} = \max\{\Phi\left(\frac{(\frac{1}{2}\Delta - \Delta)^T b}{\sqrt{b^T \Sigma b}}\right), \Phi\left(\frac{-\frac{1}{2}\Delta^T b}{\sqrt{b^T \Sigma b}}\right)\},
\]

where \(b\) is computed based on \((\tilde{x}_i, \tilde{y}_i), i = 1, \ldots, n\).

**Method comparison.** We only need to compare (5), (6), and (7).

For the ERM, \(0 \leq a_0 \leq 2\alpha\) and

\[
b = (1 + \frac{a_0\Delta^T\Sigma^{-1}\Delta_0}{1 + a_0\Delta_0^T\Sigma^{-1}\Delta_0})\Sigma^{-1}\Delta - \frac{a_0\Delta^T\Sigma^{-1}\Delta_0}{1 + a_0\Delta_0^T\Sigma^{-1}\Delta_0}\Sigma^{-1}\Delta
\]
\[
\propto \Sigma^{-1}\Delta - \frac{a_0\Delta^T\Sigma^{-1}\Delta_0}{1 + a_0\Delta_0^T\Sigma^{-1}\Delta_0}\Sigma^{-1}\Delta
\]
\[
\propto \Sigma^{-1}\Delta - \frac{a_0\Delta^T\Sigma^{-1}\Delta_0}{1 + a_0\Delta_0^T\Sigma^{-1}\Delta_0}\Sigma^{-1}\Delta.
\]
Let $c_0 = \frac{a_0 \Delta^T \Sigma^{-1} \Delta_0}{1 + a_0 \Delta^T \Sigma^{-1} \Delta_0}$ and $c_1 = |c_0| \|\Delta\|_\Sigma / \|\tilde{\Delta}\|_\Sigma$. For simplicity, let $\|v\|_\Sigma = v^T \Sigma^{-1} v$. We first lower bound it via

$$
cor(b_{\text{ERM}}, \Delta) = \frac{b^T \Delta}{\|\Delta\|_\Sigma \sqrt{b^T \Sigma b}} = \frac{\Delta^T \Sigma^{-1} \Delta - c_0 \Delta^T \Sigma^{-1} \Delta}{\|\Delta\|_\Sigma \sqrt{b^T \Sigma b}}
\geq \frac{\Delta^T \Sigma^{-1} \Delta}{\|\Delta\|_\Sigma (\|\Delta\|_\Sigma + |c_0| \|\Delta\|_\Sigma)} - \frac{|c_0 \Delta^T \Sigma^{-1} \Delta|}{\|\Delta\|_\Sigma \sqrt{b^T \Sigma b}}
\geq \frac{1}{1 + |c_0| \|\Delta\|_\Sigma / \|\Delta\|_\Sigma} - \frac{c_0 \xi \|\Delta\|_\Sigma}{\|\Delta\|_\Sigma - c_0 \|\Delta\|_\Sigma}
\geq \frac{1 - (1 + \xi) c_1 - c_1^2}{1 - c_1} = 1 - C\alpha.
$$

Similarly, we have

$$
cor(b_{\text{ERM}}, \Delta) = \frac{b^T \Delta}{\|\Delta\|_\Sigma \sqrt{b^T \Sigma b}} = \frac{\Delta^T \Sigma^{-1} \Delta - c_0 \Delta^T \Sigma^{-1} \Delta}{\|\Delta\|_\Sigma \sqrt{b^T \Sigma b}}
\leq \frac{\Delta^T \Sigma^{-1} \Delta}{\|\Delta\|_\Sigma (\|\Delta\|_\Sigma + |c_0| \|\Delta\|_\Sigma)} + \frac{|c_0 \Delta^T \Sigma^{-1} \Delta|}{(\|\Delta\|_\Sigma - c_0 \|\Delta\|_\Sigma) \|\Delta\|_\Sigma}
\leq \frac{1}{1 + c_0 \|\Delta\|_\Sigma / \|\Delta\|_\Sigma} \frac{c_0 \|\Delta\|_\Sigma}{\|\Delta\|_\Sigma - c_0 \|\Delta\|_\Sigma}
\leq \frac{\xi}{1 + c_1} \|\Delta\|_\Sigma.
$$

Hence,

$$E_{\text{ERM}}^{(u,s,t)} \geq \max \left\{ \Phi((\frac{1}{2} - C\alpha) \|\tilde{\Delta}\|_\Sigma - (\xi - C\alpha) \|\Delta\|_\Sigma), \Phi((\frac{1}{2} + C\alpha) \|\tilde{\Delta}\|_\Sigma) \right\}$$

for some constant $C$ depending on the true parameters.

For method LISA-I, using the fact that $\Delta_0 = \Delta - \tilde{\Delta}$,

$$b_{l_1} \propto (1 - \frac{a_{l_1} \Delta^T \Sigma^{-1} \Delta_0}{1 + a_{l_1} \Delta^T \Sigma^{-1} \Delta_0}) \Sigma^{-1} \Delta + \frac{a_{l_1} \Delta^T \Sigma^{-1} \Delta_0}{1 + a_{l_1} \Delta^T \Sigma^{-1} \Delta_0} \Sigma^{-1} \tilde{\Delta}
\propto \Sigma^{-1} \tilde{\Delta} + c_{l_1} \Sigma^{-1} \Delta
$$

for

$$c_{l_1} = \frac{1 + a_{l_1} \Delta^T \Sigma^{-1} \Delta_0}{a_{l_1} \Delta^T \Sigma^{-1} \Delta_0}.
$$

Hence,

$$cor(b_{l_1}, \tilde{\Delta}) = \frac{\tilde{\Delta}^T b_{l_1}}{\|\Delta\|_\Sigma \sqrt{b_{l_1}^T \Sigma b_{l_1}}} = \frac{\|\tilde{\Delta}\|_\Sigma + c_{l_1} \xi \|\Delta\|_\Sigma}{\|\Delta\|_\Sigma + c_{l_1} \|\Delta\|_\Sigma}
$$

$$cor(b_{l_1}, \Delta) = \frac{b_{l_1}^T \Delta}{\|\Delta\|_\Sigma \sqrt{b_{l_1}^T \Sigma b_{l_1}}} = \frac{\xi \|\tilde{\Delta}\|_\Sigma + c_{l_1} \|\Delta\|_\Sigma}{\|\Delta\|_\Sigma + c_{l_1} \|\Delta\|_\Sigma}.
$$

To have $E_{l_1}^{(u,s,t)} \leq E_{\text{ERM}}^{(u,s,t)}$, it suffices to require that $(\frac{1}{2} - C\alpha) \|\tilde{\Delta}\|_\Sigma \leq (\frac{1}{2} - C\alpha) \|\Delta\|_\Sigma - (\xi + C\alpha) \|\Delta\|_\Sigma$

$$\frac{1}{2} cor(b_{l_1}, \tilde{\Delta}) \|\tilde{\Delta}\|_\Sigma - cor(b_{l_1}, \Delta) \|\Delta\|_\Sigma \leq (\frac{1}{2} - C\alpha) \|\tilde{\Delta}\|_\Sigma - (\xi + C\alpha) \|\Delta\|_\Sigma
$$

$$- \frac{1}{2} cor(b_{l_1}, \tilde{\Delta}) \|\tilde{\Delta}\|_\Sigma \leq (\frac{1}{2} - C\alpha) \|\tilde{\Delta}\|_\Sigma - (\xi + C\alpha) \|\Delta\|_\Sigma.
$$
A sufficient condition is
\[ \xi \leq \left( \frac{1}{2} + \frac{1}{2} \text{cor}(b_{L1}, \tilde{\Delta}) \right) \frac{\|\tilde{\Delta}\|_\Sigma}{\|\Delta\|_\Sigma} - C\alpha, \quad \text{cor}(b_{L1}, \Delta) \geq \xi + C\alpha, \quad \text{cor}(b_{L1}, \tilde{\Delta}) \leq 1 - 2C\alpha. \]

We can find that a further sufficient condition is
\[ \xi \leq \frac{\|\tilde{\Delta}\|_\Sigma}{\|\Delta\|_\Sigma} - C\alpha, \quad \text{cor}(b_{L1}, \Delta) \leq \xi + C\alpha, \quad \text{cor}(b_{L1}, \tilde{\Delta}) \leq 1 - 2C\alpha. \] (9)

Hence, it suffices to guarantee that
\[ \|\tilde{\Delta}\|_\Sigma - \|\Delta\|_\Sigma - \epsilon_1 \alpha \]
\[ \xi \leq \frac{\|\tilde{\Delta}\|_\Sigma}{\|\Delta\|_\Sigma} - C\alpha. \] (10)

We first find sufficient conditions for the statements in (9) and (10). Parameterizing \( t = c_{L1}\|\Delta\|_\Sigma/\|\tilde{\Delta}\|_\Sigma \), we further simplify the condition in (9) and (10) as
\[ \xi \leq \frac{\|\tilde{\Delta}\|_\Sigma}{\|\Delta\|_\Sigma} - C\alpha, \quad t > 0, \quad - \frac{t}{2} \leq \xi \leq t \]
\[ \xi \leq \frac{\sqrt{1 + t^2 + 2t\xi} - 1}{t} - \epsilon_1 \alpha, \quad \xi \leq \frac{1 + \sqrt{1 + t^2 + 2t\xi}}{t + 2\xi} - \epsilon_1 \alpha. \]

We only need to require
\[ t \geq 2 \text{ and } \xi \leq \min\{\|\tilde{\Delta}\|_\Sigma/\|\Delta\|_\Sigma, 1\} - C\alpha. \]

Some tedious calculation shows that \( t \geq 2 \) can be guaranteed by
\[ \frac{1}{2} \|\Delta\|_\Sigma \leq \|\tilde{\Delta}\|_\Sigma \leq \frac{1}{\sqrt{3}\beta_{L1}} \text{ or } \frac{1}{2} \|\Delta\|_\Sigma \leq \|\tilde{\Delta}\|_\Sigma. \]

It is left to consider the constraint in (11). Notice that it holds for any \( \xi \leq 0 \). When \( \xi > 0 \), we can see
\[ \text{cor}(b_{L1}, \tilde{\Delta}) = \frac{\|\tilde{\Delta}\|_\Sigma + \xi c_{L1} \|\Delta\|_\Sigma}{\|\Delta + c_{L1} \tilde{\Delta}\|_\Sigma} = \frac{1 + t\xi}{\sqrt{1 + t^2 + 2t\xi}} \geq \frac{1 + t\xi}{1 + t} \geq \xi. \]

Hence, it suffices to guarantee that
\[ (1 - \frac{1}{2} \frac{\|\tilde{\Delta}\|_\Sigma}{\|\Delta\|_\Sigma}) \xi < \frac{1}{2} \frac{\|\tilde{\Delta}\|_\Sigma}{\|\Delta\|_\Sigma} - C\alpha. \]

If \( \|\tilde{\Delta}\|_\Sigma/\|\Delta\|_\Sigma \geq 2 \), then LHS is negative and it holds. If \( 1 \leq \|\tilde{\Delta}\|_\Sigma/\|\Delta\|_\Sigma < 2 \), then the inequality becomes \( \xi \leq 1 - C\alpha \). If \( \|\tilde{\Delta}\|_\Sigma/\|\Delta\|_\Sigma < 1 \), then the inequality becomes \( \xi \leq \frac{\|\tilde{\Delta}\|_\Sigma}{\|\Delta\|_\Sigma} - C\alpha \). Because we have required \( \xi \leq \min\{\|\tilde{\Delta}\|_\Sigma/\|\Delta\|_\Sigma, 1\} - C\alpha \) for some large enough \( C \), the constraint (11) always holds. To summarize, \( E_{L1} \leq E_{\text{ERM}} \) given that \( \xi \leq \min\{\|\tilde{\Delta}\|_\Sigma/\|\Delta\|_\Sigma, 1\} - C\alpha \) for some large enough \( C \) and \( \|\tilde{\Delta}\|_\Sigma \leq \frac{1}{\sqrt{3}\beta_{L1}} \).

For method LISA-II, we can similarly show that \( E_{L2} \leq E_{\text{ERM}} \) given that \( \xi \leq \min\{\|\tilde{\Delta}\|_\Sigma/\|\Delta\|_\Sigma, 1\} - C\alpha \) for some large enough \( C \) and \( \|\tilde{\Delta}\|_\Sigma \leq \frac{1}{\sqrt{3}\beta_{L2}} \).

B.5 Finite sample analysis

The empirical loss can be written as
\[ \mathbb{P}(1((x_i^G)^T \hat{b} + \hat{b}_0 > \frac{1}{2}) \neq y_i^G) \]
\[ = \frac{1}{2} \mathbb{P}((x_i^G)^T \hat{b} + \hat{b}_0 > \frac{1}{2}|y_i^G = 0) + \frac{1}{2} \mathbb{P}((x_i^G)^T \hat{b} + \hat{b}_0 < \frac{1}{2}|y_i^G = 1), \]

(12)
where
\[
\mathbb{P}(\left(x_i^G\right)^T \hat{b} + \hat{b}_0 > \frac{1}{2}\left|y_i^G\right| = 0) = \Phi(-\frac{1}{2} - (\mu^{(0,G)})^T \hat{b} - \hat{b}_0) / \sqrt{\hat{b}^T \hat{\Sigma} b},
\]
\[
\mathbb{P}(\left(x_i^G\right)^T \hat{b} + \hat{b}_0 < \frac{1}{2}\left|y_i^G\right| = 1) = \Phi(\frac{1}{2} - (\mu^{(1,G)})^T \hat{b} - \hat{b}_0) / \sqrt{\hat{b}^T \hat{\Sigma} b}.
\]

First notice that
\[
\hat{b}_0 = \bar{y} - \bar{x}^T \hat{b}.
\]

We have
\[
(\mu^{(0,G)})^T \hat{b} + \hat{b}_0 = (\mu^{(0,G)} - \bar{x})^T \hat{b} + \bar{y}
\]
\[
= (\mu^{(0,G)} - \mathbb{E}[x_i])^T \hat{b} + \frac{1}{2} + \{(\bar{y} - \bar{x}^T \hat{b} - (\mathbb{E}[y_i] - \mathbb{E}[x_i]^T \hat{b})\}
\]
\[
(\mu^{(1,G)})^T \hat{b} + \hat{b}_0 = (\mu^{(1,G)} - \bar{x})^T \hat{b} + \bar{y}
\]
\[
= \Delta^T \hat{b} + (\mu^{(0,G)} - \mathbb{E}[x_i])^T \hat{b} + \frac{1}{2} + R_1.
\]

Therefore, according to \[12\],
\[
\frac{1}{2} \Phi(-\frac{1}{2} - (\mu^{(0,G)})^T \hat{b} - \hat{b}_0) / \sqrt{\hat{b}^T \hat{\Sigma} b} + \frac{1}{2} \Phi(-\frac{1}{2} - (\mu^{(1,G)})^T \hat{b} - \hat{b}_0) / \sqrt{\hat{b}^T \hat{\Sigma} b}
\]
\[
= \frac{1}{2} \Phi(\frac{\theta^{(G)}^T \hat{b} + R_1}{\sqrt{\hat{b}^T \hat{\Sigma} b}}) + \frac{1}{2} \Phi(-\frac{\Delta + (\theta^{(G)})^T \hat{b} + R_1}{\sqrt{\hat{b}^T \hat{\Sigma} b}})
\]
\[
= \frac{1}{2} \Phi(\frac{\theta^{(G)}^T \hat{b} + R_1}{\sqrt{\hat{b}^T \hat{\Sigma} b}}) + \frac{1}{2} \Phi(-\frac{(\theta^{(G)})^T \hat{b} + R_1}{\sqrt{\hat{b}^T \hat{\Sigma} b}})
\]
\[
- \left\{ \frac{1}{2} \Phi(-\frac{(\theta^{(G)})^T \hat{b} + R_1}{\sqrt{\hat{b}^T \hat{\Sigma} b}}) - \frac{1}{2} \Phi(-\frac{\Delta + (\theta^{(G)})^T \hat{b} + R_1}{\sqrt{\hat{b}^T \hat{\Sigma} b}}) \right\}
\]
\[
= \frac{1}{2} - \left\{ \frac{1}{2} \Phi(-\frac{(\theta^{(G)})^T \hat{b} + R_1}{\sqrt{\hat{b}^T \hat{\Sigma} b}}) - \frac{1}{2} \Phi(-\frac{\Delta^T \hat{b} + (\theta^{(G)})^T \hat{b} + R_1}{\sqrt{\hat{b}^T \hat{\Sigma} b}}) \right\}.
\]

Then the mis-classification error can be written as
\[
\frac{1}{2} - \frac{1}{2} \left\{ \Phi(\frac{(\theta^{(G)})^T \hat{b} + R_1}{\sqrt{\hat{b}^T \hat{\Sigma} b}}) - \Phi(\frac{(\theta^{(G)})^T \hat{b} - \Delta^T \hat{b} + R_1}{\sqrt{\hat{b}^T \hat{\Sigma} b}}) \right\}.  \tag{13}
\]

Larger the \( \hat{L}(\hat{b}) \), smaller the mis-classification error.

We first find that
\[
\hat{L}(\hat{b}) - L(b) \leq C \left| \frac{(\theta^{(G)})^T \hat{b} + R_1}{\sqrt{\hat{b}^T \hat{\Sigma} b}} - \frac{(\theta^{(G)})^T b}{\sqrt{b^T \Sigma b}} \right| + C \left| \frac{(\theta^{(G)})^T \hat{b} - \Delta^T \hat{b} + R_1}{\sqrt{\hat{b}^T \hat{\Sigma} b}} - \frac{(\theta^{(G)})^T b - \Delta^T b}{\sqrt{b^T \Sigma b}} \right|
\]

In the event that
\[
\|\Sigma^{1/2}(\hat{b} - b)\|_2 = o(1) \quad \text{max}_{y,d} \|\mu^{(y,d)}\|_2 \leq C, \; \Sigma \text{ is positive definite}.
\]
for the denominator, we have
\[ |b^T \Sigma b - \hat{b}^T \Sigma \hat{b}| \leq (2\|\Sigma^{1/2}b\|_2 + \|\Sigma^{1/2}(\hat{b} - b)\|_2)\|\Sigma^{1/2}(\hat{b} - b)\|_2 \]
\[ \leq 2(1 + o(1))\|\Sigma^{1/2}b\|_2\|\Sigma^{1/2}(\hat{b} - b)\|_2 \]
\[ |\sqrt{b^T \Sigma b} - \sqrt{\hat{b}^T \Sigma \hat{b}|} \leq \frac{|\hat{b}^T \Sigma \hat{b} - b^T \Sigma b|}{\sqrt{b^T \Sigma b} + \sqrt{\hat{b}^T \Sigma \hat{b}}} \]
\[ \leq 2(1 + o(1))\|\Sigma^{1/2}(\hat{b} - b)\|_2. \]

For the numerator, we have
\[ \left| \frac{1}{2} \Delta^T \hat{b} + R_1 - \frac{1}{2} \Delta^T b \right| \leq |R_1| + \frac{1}{2}\|\Sigma^{1/2}(\hat{b} - b)\|_2. \]

We arrive at
\[ T_1 \leq (1 + o(1)) |R_1| + \frac{1}{2}\|\Sigma^{1/2}(\hat{b} - b)\|_2, \]

\[ + (1 + o(1)) \frac{1}{2}\|\Delta^T b - \Delta^T \hat{b}\|_2. \]

Moreover, \( R_1 \leq \|\hat{b} - b\|_2 + O_P\left(\frac{1}{\sqrt{n}}\right). \) To summarize,
\[ |\hat{L}(\hat{b}) - L(b)| \leq (1 + o(1))\|\hat{b} - b\|_2 + \frac{1}{\sqrt{n}}. \]

In the following, we will upper bound \( \|\hat{b} - b\|_2 \) for each method. For the **ERM method**,\n\[ \hat{b} = \{(X - \hat{X})^T(X - \hat{X})\}^{-1}(X - \hat{X})^T(y - \hat{y}). \]

It is easy to show that
\[ \|\hat{b} - b\|_2^2 = O_P\left(\frac{\sum_{i=1}^n \text{var}(y_i|x_i)}{N^2}\right) = O_P\left(\frac{p}{N}\right). \]

For the **vanilla mixup method**, we first see that
\[ \frac{1}{n} \sum_{i=1}^n \tilde{x}_i = \frac{1}{n} \sum_{i=1}^n (\lambda_i x_{i_1} + (1 - \lambda_i) x_{i_2}) = \bar{x} + O_P(n^{-1/2}) = \mu + O_P(n^{-1/2}) \]
\[ \frac{1}{n} \sum_{i=1}^n \tilde{y}_i = \pi(1) + O_P(n^{-1/2}). \]

Next,
\[ \frac{1}{n} \sum_{i=1}^n \tilde{x}_i \tilde{y}_i = \frac{1}{n} \sum_{i=1}^n \{\lambda_i^2 x_{i_1} y_{i_1} + (1 - \lambda_i)^2 x_{i_2} y_{i_2} + \lambda_i(1 - \lambda_i)x_{i_1} y_{i_2} + \lambda_i(1 - \lambda_i)x_{i_2} y_{i_1}\} \]
\[ \frac{1}{n} \sum_{i=1}^n \tilde{x}_i \tilde{y}_i - \mathbb{E}[\tilde{x}_i \tilde{y}_i] = \frac{1}{n} \sum_{i=1}^n \tilde{x}_i \tilde{y}_i - \mathbb{E}[\tilde{x}_i \tilde{y}_i|X, y] + \mathbb{E}[\tilde{x}_i \tilde{y}_i|X, y] - \mathbb{E}[\tilde{x}_i \tilde{y}_i]. \]

For \( E_2 \),
\[ E_2 = \frac{2\mathbb{E}[\lambda_i^2]}{n} \sum_{i=1}^n x_{i_1} y_i - \mathbb{E}[\tilde{x}_i \tilde{y}_i] = 2\mathbb{E}[\lambda_i^2]\mathbb{E}[x_{i_1} y_i]. \]
Hence,

\[ \|E_2\|^2 = O_P\left(\frac{p}{n}\right). \]

For \(E_1\), conditioning on \((X, y)\), \(\lambda^2 x_i, y_i - \frac{E[\lambda^2]}{n} \sum_{i=1}^{n} x_i y_i\) are independent sub-Gaussian vectors. The sub-Gaussian norm of \(\frac{1}{n} \sum_{i=1}^{n} \lambda^2 x_i, y_i - \frac{E[\lambda^2]}{n} \sum_{i=1}^{n} x_i y_i\) (conditioning on \((X, y)\)) can be upper bounded by \(c \max_{i \leq N} |x_{i,j}| / \sqrt{n}\). Hence

\[ \mathbb{P}\left(\|E_1\|_2 \geq t|x, y\right) = 2 \exp\left\{-\frac{c_2 n t^2}{\max_p \max_{j=1}^p \max_{i \leq N} x_{i,j}^2}\right\}. \]

As \(x_{i,j}\) are Gaussian distributed, we know that

\[ \mathbb{P}\left(\sum_{j=1}^{p} \max_{j \leq n} x_{i,j}^2 \geq p \log n\right) \leq \exp\{-c_3 \log n\}. \]

Hence, with probability at least \(1 - \exp(-c_1 \log n)\),

\[ E_1 \leq \frac{C p \log n}{n}. \]

To summarize,

\[ \left\|\frac{1}{n} \sum_{i=1}^{n} \hat{x}_i \hat{y}_i - \left(\frac{1}{n} \sum_{i=1}^{n} \hat{x}_i\right)\left(\frac{1}{n} \sum_{i=1}^{n} \hat{y}_i\right) - \text{cov}(\hat{x}_i, \hat{y}_i)\right\|_2^2 = O_P\left(\frac{p \log n}{n}\right). \]

Similarly, we can show that

\[ \left\|\frac{1}{n} \sum_{i=1}^{n} \hat{x}_i \hat{x}_i^T - \left(\frac{1}{n} \sum_{i=1}^{n} \hat{x}_i\right)\left(\frac{1}{n} \sum_{i=1}^{n} \hat{x}_i\right)^T - \text{cov}(\hat{x}_i)\right\|_2^2 = O_P\left(\frac{p \log n}{n}\right). \]

Hence,

\[ \|\hat{b} - b\|^2 = O_P\left(\frac{p \log n}{n}\right). \]

For the LISA-I, we first see that

\[ \frac{1}{n} \sum_{i=1}^{n} x_i^{(1)} = \frac{1}{n} \sum_{y_i=1} \left(\lambda_i x_i^{(1,G)} + (1 - \lambda_i) x_i^{(1,B)}\right) + \frac{1}{n} \sum_{y_i=0} \left(\lambda_i x_i^{(G,0)} + (1 - \lambda_i) x_i^{(B,0)}\right) \]

\[ = \frac{1}{2} \left(\bar{x}^{(1,G)} + \bar{x}^{(1,B)}\right) \tilde{\pi}_1 + \frac{1}{2} \left(\bar{x}^{(0,G)} + \bar{x}^{(0,B)}\right) \tilde{\pi}_0. \]

We have

\[ \frac{1}{n} (X^{(1)})^T y - \bar{y} - \frac{1}{n} \sum_{i=1}^{n} x_i^{(1)} - \text{cov}(x_i^{(1)}, y_i) = \frac{1}{n} (X^{(1)})^T y - \bar{y} - \frac{1}{n} \sum_{i=1}^{n} x_i^{(1)} - \text{cov}(x_i^{(1)}, y_i) \]

\[ + \text{cov}(x_i^{(1)}, y_i|X, y) \]

\[ E_1 \]

\[ + \text{cov}(x_i^{(1)}, y_i) \]

\[ E_2 \]

\[ \]

\[ E_2 = \frac{1}{2} \left(\bar{x}^{(1,G)} + \bar{x}^{(1,B)}\right) \tilde{\pi}_1 + \frac{1}{2} \left(\bar{x}^{(0,G)} + \bar{x}^{(0,B)}\right) \tilde{\pi}_0) - \text{cov}(x_i^{(1)}, y_i) \]

\[ = \frac{1}{2} \left(\bar{x}^{(1,G)} + \bar{x}^{(1,B)} - \bar{x}^{(0,G)} - \bar{x}^{(0,B)}\right) \tilde{\pi}_1 \tilde{\pi}_0 - \pi^{(1)} \pi^{(0)} \Delta. \]

It is easy to show that

\[ \|E_2\|^2 = O_P\left(\frac{p}{\min_{y,e} n(y,e)}\right). \]
For $E_1$, conditioning on $X$ and $y$, $x_i^{(\lambda)} y_i - \mathbb{E}[x_i^{(\lambda)} y_i | X, y]$ are independent sub-Gaussian vectors with mean zero. The sub-Gaussian norm of $\frac{1}{n} \sum_{i=1}^{n} x_i^{(\lambda)} y_i$ (conditioning on $X$ and $y$) can be upper bounded by $c \max_{i \leq n} |x_{i,j}| / \sqrt{N}$.

$$\mathbb{P}(\|E_1\|_2 \geq t | X, y) = \mathbb{P}\left( \left\| \frac{1}{n} \sum_{i=1}^{n} \{x_i^{(\lambda)} y_i - \mathbb{E}[x_i^{(\lambda)} y_i | X, y]\} \|_2 \geq t \right\| | X, y \right) \leq 2 \exp \left\{ - \frac{c^2 n t^2}{\sum_{j=1}^{p} \max_{i \leq n} x_{i,j}^2} \right\}.$$

Hence,

$$E_1 = O_P(\sqrt{\frac{\sum_{j=1}^{p} \max_{i \leq n} x_{i,j}^2}{n}}) = O_P\left( \frac{p \log n}{n} \right).$$

To summarize,

$$\| \frac{1}{n} (X^{(\lambda)})^T y - \mathbb{E}[x_i^{(\lambda)} y_i]\|_2^2 = O_P\left( \frac{p}{\min_{y,c} n(y,c)} + \frac{p \log n}{n} \right).$$

We can use similar analysis to bound

$$\| \frac{1}{N} (X^{(\lambda)})^T X^{(\lambda)} - \mathbb{E}[x_i^{(\lambda)} (x_i^{(\lambda)})^T]\|_2.$$

The sub-exponential norm of $\frac{1}{N} \sum_{i=1}^{N} x_i^{(\lambda)} x_i^{(\lambda)} (\text{conditioning on } X)$ can be upper bounded by $\max_{i \leq N} |x_{i,j}| / \sqrt{N}$. We can show that

$$\| \frac{1}{n} (X^{(\lambda)})^T X^{(\lambda)} - \mathbb{E}[x_i^{(\lambda)} (x_i^{(\lambda)})^T]\|_2 = O_P\left( \frac{p}{\min_{y,c} n(y,c)} + \frac{p \log n}{n} \right).$$

For the **LISA-II**, we first see that

$$\frac{1}{n} \sum_{i=1}^{n} \tilde{x}_i = \frac{1}{n} \sum_{D_i = G} \lambda_i x_{i1}^{(1,G)} + (1 - \lambda_i) x_{i2}^{(0,G)} + \frac{1}{n} \sum_{D_i = B} \lambda_i x_{i1}^{(1,B)} + (1 - \lambda_i) x_{i2}^{(0,B)}$$

$$= \frac{1}{2} (\tilde{x}^{(1,G)} + \tilde{x}^{(0,G)}) \tilde{\pi}_G + \frac{1}{2} (\tilde{x}^{(1,B)} + \tilde{x}^{(0,B)}) \tilde{\pi}_B$$

$$\tilde{y} = \frac{1}{2}.$$

Next,

$$\frac{1}{n} \sum_{i=1}^{n} \tilde{x}_i \tilde{y}_i = \frac{1}{n} \sum_{D_i = G} \lambda_i^2 x_{i1}^{(1,G)} + \lambda_i (1 - \lambda_i) x_{i2}^{(0,G)} + \frac{1}{n} \sum_{D_i = B} \lambda_i^2 x_{i1}^{(1,B)} + \lambda_i (1 - \lambda_i) x_{i2}^{(0,B)}$$

$$= \frac{1}{2} (\tilde{x}^{(1,G)} + \tilde{x}^{(0,G)}) \tilde{\pi}_G - \frac{1}{4} (\tilde{x}^{(1,B)} + \tilde{x}^{(0,B)}) \tilde{\pi}_B - \text{cov}(\tilde{x}_i, \tilde{y}_i)\left(\tilde{x}_i, \tilde{y}_i | X, y\right) + \text{cov}(\tilde{x}_i, \tilde{y}_i | X, y) - \text{cov}(\tilde{x}_i, \tilde{y}_i).$$

For $E_2$,

$$E_2 = \tilde{\pi}^{(G)}(\mathbb{E}[\lambda_i^2](\tilde{x}^{(1,G)} + \tilde{x}^{(0,G)}) + \frac{1}{2} \mathbb{E}[\lambda_i^2](\tilde{x}^{(1,B)} + \tilde{x}^{(0,B)}) + \frac{1}{2} \mathbb{E}[\lambda_i^2]) - \frac{1}{4} (\tilde{x}^{(1,G)} + \tilde{x}^{(0,G)}) \tilde{\pi}_G - \frac{1}{4} (\tilde{x}^{(1,B)} + \tilde{x}^{(0,B)}) \tilde{\pi}_B - \text{var}(\lambda_i) \Delta$$

$$= \tilde{\pi}^{(G)} \text{var}(\lambda_i)(\tilde{x}^{(1,G)} - \tilde{x}^{(0,G)}) + \tilde{\pi}^{(B)} \text{var}(\lambda_i)(\tilde{x}^{(1,B)} - \tilde{x}^{(0,B)}) - \text{var}(\lambda_i) \Delta.$$

Notice that $E_2$ is a sub-Gaussian vector with sub-Gaussian norm upper bounded by

$$\frac{\tilde{\pi}_G^2}{n^{(1,G)}} + \frac{\tilde{\pi}_G^2}{n^{(0,G)}} + \frac{\tilde{\pi}_B^2}{n^{(1,B)}} + \frac{\tilde{\pi}_B^2}{n^{(0,B)}} \leq \frac{4}{n} \max_{y,d} \frac{\pi_y^2}{\pi_d^2}.$$
Using sub-Gaussian concentration, we can show that
\[ E_2 = O_P\left(\sqrt{\frac{p}{n}} \max_{y,d} \hat{\pi}_{y|d}\right). \]

Notice that \( \max_{y,d} \hat{\pi}_{y|d} \geq 1. \) For \( E_1, \) conditioning on \( X \) and \( y \bar{x}_i \bar{y}_i - \mathbb{E}[\bar{x}_i \bar{y}_i|X,y] \) are independent sub-Gaussian vectors with mean zero. The sub-Gaussian norm of \( \frac{1}{n} \sum_{i=1}^n \bar{x}_i \bar{y}_i \) conditioning on \( X \) and \( y \) can be upper bounded by \( c \max_{i,j} |x_{i,j}|. \) Similar analysis on \( E_1 \) leads to
\[ \frac{1}{n} \sum_{i=1}^n \bar{x}_i \bar{y}_i - \bar{x} \bar{y} - \text{cov}(\bar{x}, \bar{y}) = O_P\left(\sqrt{\frac{p \log n}{n}} + \sqrt{\frac{p}{n}} \max_{y,d} \hat{\pi}_{y|d}\right). \]

For the sample covariance matrix, we can also show that
\[ \left\| \frac{1}{n} \sum_{i=1}^n \hat{x}_i \hat{x}_i^T - \left( \frac{1}{n} \sum_{i=1}^n \hat{x}_i \right) \left( \frac{1}{n} \sum_{i=1}^n \hat{x}_i \right)^T - \text{cov}(\hat{x}_i) \right\|_2 = O_P\left(\sqrt{\frac{p \log n}{n}} + \sqrt{\frac{p}{n}} \max_{y,d} \hat{\pi}_{y|d}\right). \]

### B.6 Domain Shifts: Proof of Theorem 2

It still holds that \( \bar{\Delta}^* = 2(\mu(0,\star) - \mathbb{E}[x_i^{(1)}]) = 2(\mu(0,\star) - \mathbb{E}[x_i]). \) It is easy to show that the worst group mis-classification error for this new environment is
\[ E_A^{(wst,\star)} = \max \left\{ F \left( \frac{\bar{\Delta}^*}{\sqrt{\text{cov}(\bar{\Delta}^*)}} \right), \text{F} \left( \frac{\bar{\Delta}^*}{\sqrt{\text{cov}(\bar{\Delta}^*)}} \right) \right\}, \]
where \( A \in \{ \text{ERM, mix, L1, L2} \}. \) Notice that
\[ \bar{\Delta}^* = 2\mu_{\{0,\star\}} - (\mu_{\{0,G\}} + \mu_{\{1,B\}}) = \bar{\Delta} + \mu_{\{0,\star\}} - \mu_{\{0,G\}} \]
We assume \( \|\bar{\Delta}^*\|_2 = \|\bar{\Delta}\|_2. \) Let \( \xi^* = \text{cor}(\Delta, \bar{\Delta}^*) \) and \( \gamma = \text{cor}(\bar{\Delta}, \bar{\Delta}^*). \) We have
\[ \text{cor}(b_{\text{ERM}, \bar{\Delta}^*}) = \frac{\gamma \|\bar{\Delta}\|_{\Sigma} \|\bar{\Delta}^*\|_{\Sigma} - c_0 \xi^* \|\bar{\Delta}\|_{\Sigma} \|\bar{\Delta}^*\|_{\Sigma}}{\|\bar{\Delta}^*\|_{\Sigma} \|\bar{\Delta}\|_{\Sigma} + c_0 \Delta \|\bar{\Delta}^*\|_{\Sigma}} \]
\[ = \frac{\gamma \|\bar{\Delta}\|_{\Sigma} - c_0 \xi^* \|\Delta\|_{\Sigma}}{\|\bar{\Delta}^*\|_{\Sigma} + c_0 \Delta \|\bar{\Delta}\|_{\Sigma}} = \gamma \pm C \alpha. \]

Hence,
\[ E_{\text{ERM}}^{(wst)} \geq \max \left\{ F((\frac{\gamma}{2} - C \alpha) \|\bar{\Delta}\|_{\Sigma} - (\xi - C \alpha) \|\Delta\|_{\Sigma}), F((-\frac{\gamma}{2} - C \alpha) \|\bar{\Delta}\|_{\Sigma}) \right\} \]
for some constant \( C \) depending on the true parameters.

Hence,
\[ \text{cor}(b_{L1, \bar{\Delta}^*}) = \frac{(\bar{\Delta}^*_*) b_{L1}}{\|\bar{\Delta}^*\|_{\Sigma} \sqrt{\text{cov}(\bar{\Delta}^*_*)}} = \frac{\gamma \|\bar{\Delta}\|_{\Sigma} + c_{L1} \xi^* \|\Delta\|_{\Sigma}}{\|\Delta + c_{L1} \Delta\|_{\Sigma}}. \]

To have \( E_{L1}^{(wst,\star)} \leq E_{\text{ERM}}^{(wst,\star)} \), it suffices to require that \( (\frac{\gamma}{2} - C \alpha) \|\bar{\Delta}\|_{\Sigma} \leq (\frac{\gamma}{2} - C \alpha) \|\bar{\Delta}\|_{\Sigma} - (\xi + C \alpha) \|\Delta\|_{\Sigma} \)

\[ \frac{1}{2} \text{cor}(b_{L1, \bar{\Delta}^*}) \|\bar{\Delta}\|_{\Sigma} - \text{cor}(b_{L1, \Delta}) \|\Delta\|_{\Sigma} \leq \left( \frac{\gamma}{2} - C \alpha \right) \|\bar{\Delta}\|_{\Sigma} - (\xi + C \alpha) \|\Delta\|_{\Sigma} \]

\[ - \frac{1}{2} \text{cor}(b_{L1, \bar{\Delta}^*}) \|\bar{\Delta}\|_{\Sigma} \leq \left( \frac{\gamma}{2} - C \alpha \right) \|\Delta\|_{\Sigma} - (\xi + C \alpha) \|\Delta\|_{\Sigma}. \]

A sufficient condition is
\[ \xi \leq \left( \frac{\gamma}{2} + \frac{1}{2} \text{cor}(b_{L1, \bar{\Delta}^*}) \right) \|\bar{\Delta}\|_{\Sigma} - C \alpha, \text{cor}(b_{L1, \Delta}) \leq \xi + C \alpha, \text{cor}(b_{L1, \bar{\Delta}^*}) \leq \gamma - 2 C \alpha. \]
We can find that a further sufficient condition is
\[
\xi \leq \frac{1 + \gamma \|\tilde{\Delta}\|_\Sigma}{2 \|\Delta\|_\Sigma} - C\alpha, \quad \xi^* \leq \frac{\gamma(\|\tilde{\Delta} + c_{l,1}\Delta\|_\Sigma - \|\tilde{\Delta}\|_\Sigma)}{c_{l,1}\|\Delta\|_\Sigma} - \epsilon_1 \alpha \tag{16}
\]
\[
\|\tilde{\Delta} + c_{l,1}\Delta\|_\Sigma \geq \|\tilde{\Delta}\|_\Sigma, \quad \xi \leq \frac{c_{l,1}\|\Delta\|_\Sigma}{\|\tilde{\Delta} + c_{l,1}\Delta\|_\Sigma - \|\Delta\|_\Sigma} - \epsilon_1 \alpha \tag{17}
\]
\[
\xi \leq \left(\frac{\gamma}{2} + \frac{1}{2} cor(b_{l,1}, \tilde{\Delta}^*)\right) \frac{\|\tilde{\Delta}\|_\Sigma}{\|\Delta\|_\Sigma} - C\alpha. \tag{18}
\]
We first find sufficient conditions for the statements in (9) and (10). Parameterizing \(t = c_{l,1}\|\Delta\|_\Sigma/\|\tilde{\Delta}\|_\Sigma\), we further simplify the condition in (16) and (17) as
\[
\xi \leq \frac{1 + \gamma}{2} \frac{\|\tilde{\Delta}\|_\Sigma}{\|\Delta\|_\Sigma} - C\alpha, \quad t > 0, \quad \xi^* \leq \frac{\gamma(\sqrt{1 + t^2 + 2t\xi} - 1)}{t} - \epsilon_1 \alpha,
\]
\[-t \leq \xi \leq t, \quad \xi \leq \frac{1 + \sqrt{1 + t^2 + 2t\xi}}{t + 2\xi} - \epsilon_1 \alpha.
\]
We only need to require
\[t \geq 2 \text{ and } \xi \leq \min\left\{\frac{1 + \gamma}{2} \frac{\|\tilde{\Delta}\|_\Sigma}{\|\Delta\|_\Sigma}, 1\right\} - C\alpha, \quad \xi^* \leq \gamma \xi.\]
Some tedious calculation shows that \(t \geq 2\) can be guaranteed by
\[\frac{1}{2} \|\Delta\|_\Sigma \leq \|\tilde{\Delta}\|_\Sigma \leq \frac{1}{\sqrt{3c_{l,1}}} \text{ or } \frac{1}{2} \|\Delta\|_\Sigma \geq \|\tilde{\Delta}\|_\Sigma.\]
It is left to consider the constraint in (18). Notice that it holds for any \(\xi \leq 0\). When \(\xi > 0\), we can see
\[cor(b_{l,1}, \tilde{\Delta}^*) = \frac{\gamma\|\tilde{\Delta}\|_\Sigma + \xi^*c_{l,1}\|\Delta\|_\Sigma}{\|\Delta + c_{l,1}\Delta\|_\Sigma} = \frac{\gamma + t\xi^*}{\sqrt{1 + t^2 + 2t\xi}} \geq \frac{\gamma + t\xi^*}{1 + t}.
\]
Hence, it suffices to guarantee that
\[\xi^* + \gamma \geq \frac{2\|\Delta\|_\Sigma}{\|\tilde{\Delta}\|_\Sigma} \xi + C\alpha.\]
To summarize, it suffices to require
\[\|\tilde{\Delta}\|_\Sigma \leq \frac{1}{\sqrt{3c_{l,1}}}, \quad 0 \leq \xi^* \leq \gamma \xi, \quad \xi \leq \min\left\{\frac{\gamma}{2} \frac{\|\tilde{\Delta}\|_\Sigma}{\|\Delta\|_\Sigma}, 1\right\} - C\alpha.
\]
For LISA-II, we can similarly show that \(E_{l,2}^{(\text{wast})} \leq E_{\text{ERM}}^{(\text{wast})}\) given that
\[\|\tilde{\Delta}\|_\Sigma \leq \frac{1}{\sqrt{3c_{l,2}}}, \quad 0 \leq \xi^* \leq \gamma \xi, \quad \xi \leq \min\left\{\frac{\gamma}{2} \frac{\|\tilde{\Delta}\|_\Sigma}{\|\Delta\|_\Sigma}, 1\right\} - C\alpha.
\]