A APPENDIX

A.1 ADDITIONAL BACKGROUND

A.1.1 TV-STABILITY

TV Stability implies that an (ε, δ)-DP algorithm assures $(\exp(\varepsilon) - 1 + \delta)$-TV stability, Kulynych et al. (2022) prove a much tighter bound. They show that an (ε, δ)-DP algorithm implies $\delta^{'}(\sigma)$-TV stability where

$$\delta^{'}(\sigma) = \frac{\exp(\varepsilon(\sigma)) - 1 + 2\delta}{\exp(\varepsilon(\sigma)) + 1},$$

is a function of some controllable parameter σ that adjusts the stability of the algorithm. We empirically vary σ to provide different levels of stability following this definition to assess the sensitivity of SL algorithms when facing different forms of distribution shift.

A.1.2 DIFFERENTIALLY PRIVATE STOCHASTIC GRADIENT DESCENT

As mentioned in Section 3, a differentially private algorithm M maps a training dataset to a neural network set of parameters w_i. If $L(w, x, y)$ is the learning objective, given model parameters w, features x, and labels y.

In the non-private ERM setting, stochastic gradient descent (SGD) performs the following gradient update step, drawing from N samples:

$$w^{t+1} = w^t - \eta_t \frac{1}{N} \sum_{i \in N_t} (\nabla L_i(w^t, x_i, y_i))$$

where η_t is the step-size for update t, ∇ is the model gradient, and N_t is the set of examples sampled at iteration t. In the private DP-SGD setting, we make two modifications to the update step: (1) clipping the gradient in each mini-batch to a maximal norm C and (2) adding Gaussian noise to the mean of the clipped gradients. The new update step is:

$$w^{t+1} = w^t - \eta_t \frac{1}{N} \sum_{i \in N_t} (\textbf{clip}(\nabla L_i(w^t, x_i, y_i)) + N(0, \sigma^2 C^2 I))$$

where C is the the maximal clipping norm, σ is the noise multiplier, and the **clip** function is **clip**(x)=$x/\max(1, \frac{||x||_2}{C})$. In our experiments, we use σ, the noise multiplier, as our measure of stability, and treat C as a hyperparameter.

A.2 ADDITIONAL DATASET INFORMATION

A.2.1 SYNTHETIC DATASETS

`CIFAR10-C:` For a synthetic covariate shift, we use the CIFAR10-C dataset created by Hendrycks et al from CIFAR-10. The dataset was created by corrupting CIFAR-10's test set of 10,000 images with 19 types of algorithmically generated corruptions from noise, blur, weather, and digital categories of 5 different shift severities for a total of 95 different covariate shifts (Hendrycks & Dietterich, 2019).

`Imbalanced-CIFAR:` For a synthetic label shift, we induce class imbalance in CIFAR-10, overrepresenting some classes and underrepresenting others to create a shift in $P(Y)$. These shifts were created randomly, where the percentage of samples in the shifted dataset from the original test dataset was chosen randomly from $10 - 100\%$.

`Waterbirds` Object recognition models often suffer from poor worst-subgroup utility in the presence of subpopulation shift due to learning spurious correlations. We examine the efficacy of DP to prevent relying on these spurious correlations on the synthetic `Waterbirds` dataset. This is a dataset of bird images which combines the bird photographs from the Caltech-UCSD Birds-200-2011 (CUB-200-2011) dataset (Wah et al., 2011) with image backgrounds from the Places dataset (Zhou et al., 2017) using the procedure from (Sagawa et al., 2019).

A.2.2 NATURAL DATASETS

`Cell Out of Sample:` We explore naturally occurring biological shifts in the Cells-Out-of-Sample dataset (COOS), a image classification setting one of the few natural shift datasets created for the purpose of a covariate shift (Lu et al., 2019). COOS consists of 132,209 images of mouse cells of 7 biological classes, with 4 different test sets of increasing degrees of covariate shift, where some images are random subsets of the training data, while others are from experiments reproduced months later and imaged by different instruments.

`PovertyMap` As satellite imagery becomes more used in different machine learning techniques to understand demographic attributes (Xie et al., 2016; Oshri et al., 2018; de Mattos et al., 2021), it is necessary to understand how ML models perform in these settings. Recent work has shown that image models have poor worst-subgroup utility in the presence of subpopulation shifts (Sagawa et al., 2019). We use the variant of the `PovertyMap` datasets from the `WILDS` dataset to perform poverty estimation based on satellitle images from 23 African countries, from both urban and rural areas (Koh et al., 2021).

`MIMIC-III Notes` Clinical notes contain information that can often be used to predict potential mortality or long length of stays, but disparities in performance of predictive models have been reported (Chen et al., 2018). Using the MIMIC-III dataset of all clinical notes from 25,879 adult patients from Beth Israel Deaconess Medical Center (Johnson et al., 2016), we predict hospital mortality of patients in critical care. Notes were collected in the first 48 hours of an intensive care unit (ICU) stay; discharge notes were excluded. We only included patients that stayed in the ICU for more than 48 hours. We use the tf-idf statistics of the 10,000 most frequent words as features for each note per patient.

A.3 ADDITIONAL EXPERIMENTAL SETUP INFORMATION

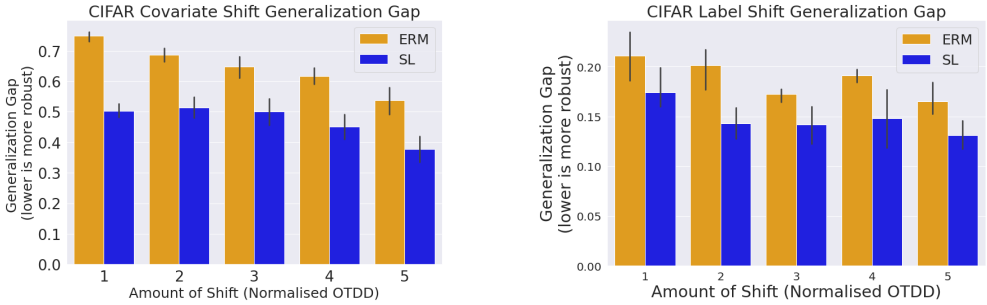
For this work, we use Nvidia Volta V100 GPU's for all experiments. To generate error bars, we ran each hyperparameter setting over three random seeds. All datasets and code are licensed under the MIT data license.

Imbalanced CIFAR and CIFAR-C For the DP models, we look at noise levels of $\{0.0001, 0.001, 0.001, 0.1, 0.5, 1.0, 1.5\}$ for each dataset. We trained each model with a for 100 epochs, early stopping if the validation loss did not decrease for more than 10 epochs. We used DP-SGD and SGD as optimizers with a Nesterov momentum 0.9 for DP and ERM experiments, respectively. For each test dataset, we perform the following hyperparameter search: learning rate = $\{0.1, 0.01, 0.001\}$, clipping norm = $\{0.1, 1, 3, 5, 10\}$, batch size = $\{16, 32, 64, 512, 1024, 2048\}$. For all CIFAR-10C and CIFAR-10 experiments, we use the state-of-the-art end-to-end CNN models defined by (Tramèr & Boneh, 2021). For all models, we use Opacus to implement DP (Yousefpour et al., 2021).

Waterbirds We defined the five shift severities as the increase in subopulation shift from training to test distribution. Specifically, the proportion of waterbirds on land in the training set is decreased in each shift: 1 (40%), 2 (30%), 3 (20%), 4 (10%), 5 (5%). In the test set the proportions are equal between the four groups of {(waterbirds, water background), (landbirds, water background), (waterbirds, land background), (landbirds, land background)}. We trained logistic regression models which were implemented in Tensorflow Privacy (McMahan & Andrew, 2018). We perform a hyperparameter search over the following grid: batch size = {16, 32, 64, 128}, learning rate = {0.001, 0.005, 0.01, 0.05, 0.1}, with a fixed clipping norm 1.0. We examine the noise multipliers of $\{0.1, 0.5, 1.0, 1.5\}$.

Natural Experiments For the datasets `COOS`, `PovertyMap`, and `MIMIC-III`, we perform a hyperparameter search of the following grid: batch size = {16, 32, 64, 128}, learning rate = {0.001, 0.005, 0.01, 0.05, 0.1} set with a clipping norm 1.0.

For the `PovertyMap` and `COOS` datasets, we end-to-end train a ResNet-18 model implemented in PyTorch with Opacus (Yousefpour et al., 2021). For the MIMIC-III notes, we train logistic re-



(a) Covariate Shift (b) Label Shift

Figure 3: On the `CIFAR10-C` and `Imbalanced-CIFAR` datasets, stable learning (SL) has a lower generalization gap than ERM across all shift types and severities. Furthermore, as the shift increases, the generalization gap drops and $\Delta_G > 0$ across different levels of shift and noise level, both of which indicate greater robustness to distribution shift than ERM.

gression models which were implemented in Tensorflow Privacy (McMahan & Andrew, 2018). For these datasets, we examine the noise multipliers of $\{0.1, 0.5, 1.0, 1.5\}$.

A.4 Additional Experimental Results

A.4.1 Covariate Shift

We see a performance degradation effect with covariate shift, with certain shifts (such as brightness, spatter, and saturate in Fig. 4). At shift$= 4$, we can see that SL outperforms ERM in terms of accuracy, indicating that at greater shifts, SL is more robust than ERM, even if ERM starts at a higher testing accuracy on the no shift dataset.

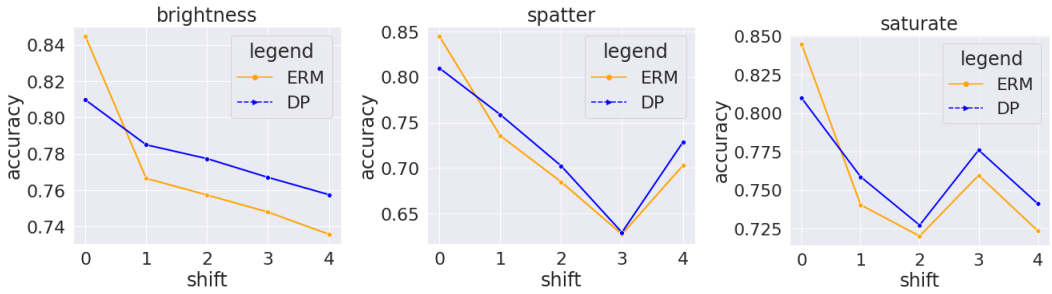


Figure 4: Comparison of DP and ERM test accuracy for 3 representative shifts at 4 different levels of shift severities, where shift$= 0$ is the nonshifted test dataset. The performance of DP models is more stable and degrades much less as the shift becomes more severe. This comparison is done at $\sigma = 0.1$

In Fig. 2a, we see that with additional noise comes additional robustness. However, this happens at a cost to accuracy. In Table 7 and Table 8, we see the full results from the higher noise multipliers 0.5 and 1.5. Under a higher noise setting, the SL models are more robust, with a greater Δ_G than $\sigma = 0.1$ in Table 2, but they suffer a greater performance drop. This displays the robustness-accuracy tradeoff seen with covariate shift. Furthermore, this indicates that the amount of stability used for robustness can be treated as a hyperparameter, where the correct noise level can find the best level of tradeoff.

A.4.2 Label Shift

In Table 10 and Table 11, we see the full results from the higher noise multipliers 0.5 and 1.5. Under a higher noise setting, the SL models are more robust, with a greater Δ_G than $\sigma = 0.1$ in Table 2.

Shift Severity	ERM train acc	ERM test acc	SL train acc	SL test acc	Accuracy Gain	$\mathbf{\Delta_G}$
None	0.938 ± 0.001	0.771 ± 0.034	0.748 ± 0.165	0.700 ± 0.113	-0.071 ± 0.113	$\mathbf{0.119} \pm 0.057$
1	0.923 ± 0.025	0.748 ± 0.042	0.567 ± 0.073	0.543 ± 0.048	-0.205 ± 0.062	$\mathbf{0.151 \pm 0.039}$
2	0.919 ± 0.0	0.686 ± 0.050	0.663 ± 0.102	0.582 ± 0.093	-0.104 ± 0.054	$\mathbf{0.152 \pm 0.059}$
3	0.892 ± 0.049	0.649 ± 0.097	0.634 ± 0.117	0.533 ± 0.121	-0.116 ± 0.065	$\mathbf{0.142 \pm 0.087}$
4	0.875 ± 0.043	0.618 ± 0.067	0.604 ± 0.122	0.489 ± 0.122	-0.128 ± 0.078	$\mathbf{0.143 \pm 0.074}$
5	0.857 ± 0.047	0.538 ± 0.115	0.559 ± 0.123	0.406 ± 0.123	-0.132 ± 0.081	$\mathbf{0.167 \pm 0.086}$

Table 7: For $\sigma = 0.5$, SL improves robustness of models to covariate shift for `CIFAR10-C`, with $\Delta_G > 0$. However we see a greater robustness-accuracy tradeoff than with $\sigma = 0.1$ (Seen in Table 2)

Shift Severity	ERM train acc	ERM test acc	SL train acc	SL test acc	Accuracy Gain	$\mathbf{\Delta_G}$
1	0.923 ± 0.025	0.748 ± 0.0420	0.494 ± 0.165	0.457 ± 0.149	-0.291 ± 0.143	$\mathbf{0.138 \pm 0.052}$
2	0.919 ± 0.000	0.686 ± 0.050	0.433 ± 0.161	0.406 ± 0.137	-0.280 ± 0.122	$\mathbf{0.206 \pm 0.062}$
3	0.892 ± 0.049	0.649 ± 0.097	0.541 ± 0.131	0.489 ± 0.132	-0.160 ± 0.106	$\mathbf{0.191 \pm 0.056}$
4	0.875 ± 0.043	0.618 ± 0.067	0.501 ± 0.156	0.419 ± 0.137	-0.199 ± 0.099	$\mathbf{0.175 \pm 0.101}$
5	0.857 ± 0.047	0.538 ± 0.115	0.606 ± 0.118	0.448 ± 0.112	-0.09 ± 0.047	$\mathbf{0.161 \pm 0.070}$

Table 8: For $\sigma = 1.5$, SL improves robustness of models to covariate shift for `CIFAR10-C`, with $\Delta_G > 0$. However we see a greater robustness-accuracy tradeoff than with $\sigma = 0.1$ (Seen in Table 2)

Shift Severity	ERM train acc	ERM test acc	SL train acc	SL test acc	Accuracy Gain	$\mathbf{\Delta_G}$
1	0.906 ± 0.032	0.697 ± 0.052	0.9 ± 0.063	0.698 ± 0.059	0.001 ± 0.017	0.007 ± 0.063
2	0.901 ± 0.046	0.681 ± 0.063	0.91 ± 0.061	0.683 ± 0.067	0.003 ± 0.019	-0.006 ± 0.092
3	0.888 ± 0.042	0.611 ± 0.084	0.884 ± 0.064	0.605 ± 0.093	-0.006 ± 0.02	-0.001 ± 0.077
4	0.871 ± 0.042	0.618 ± 0.075	0.864 ± 0.056	0.605 ± 0.094	-0.014 ± 0.041	-0.006 ± 0.079
5	0.857 ± 0.049	0.526 ± 0.12	0.862 ± 0.109	0.507 ± 0.135	-0.019 ± 0.053	-0.024 ± 0.085

Table 9: For $\sigma = 0.0001$, SL has similar results to ERM to covariate shift for `CIFAR10-C`, with $\Delta_G > 0$ not always being true.

However, with $\sigma = 0.5$, we see mixed results of performance, with some shifts with performance gains and other with drops. In $\sigma = 1.5$, we see all shifts with greater robustness, but more significant performance drop than $\sigma = 0.1$. Once again, similar to covariate shift, these results illustrate the tradeoff between robustness and accuracy.

Shift Severity	ERM train acc	ERM test acc	SL train acc	SL test acc	Accuracy Gain	Δ_G
1	0.776 ± 0.242	0.577 ± 0.174	0.803 ± 0.02	0.623 ± 0.011	**-0.046 ± 0.091**	**0.019 ± 0.052**
2	0.863 ± 0.079	0.576 ± 0.066	0.635 ± 0.151	0.509 ± 0.134	**0.067 ± 0.061**	**0.161 ± 0.038**
3	0.734 ± 0.121	0.570 ± 0.118	0.614 ± 0.287	0.491 ± 0.228	**0.079 ± 0.168**	**0.041 ± 0.054**
4	0.682 ± 0.210	0.509 ± 0.128	0.598 ± 0.19	0.451 ± 0.171	**0.058 ± 0.217**	**0.026 ± 0.080**
5	0.590 ± 0.057	0.417 ± 0.041	0.638 ± 0.258	0.522 ± 0.197	**-0.105 ± 0.047**	**0.057 ± 0.020**

Table 10: For $\sigma = 0.5$, SL improves robustness of models to label shift for `Imbalanced-CIFAR10`. However we see a significantly greater robustness-accuracy tradeoff than with $\sigma = 0.1$ in Table 6, with shifts 1 and 5 resulting in performance drops.

Shift Severity	ERM train acc	ERM test acc	SL train acc	SL test acc	Accuracy Gain	$\mathbf{\Delta_G}$
1	0.776 ± 0.242	0.577 ± 0.174	0.56 ± 0.214	0.426 ± 0.16	$\mathbf{-0.151 \pm 0.039}$	$\mathbf{0.065 \pm 0.034}$
2	0.863 ± 0.079	0.576 ± 0.066	0.612 ± 0.107	0.473 ± 0.091	$\mathbf{-0.103 \pm 0.114}$	$\mathbf{0.148 \pm 0.056}$
3	0.734 ± 0.121	0.570 ± 0.118	0.456 ± 0.238	0.364 ± 0.186	$\mathbf{-0.206 \pm 0.166}$	$\mathbf{0.072 \pm 0.024}$
4	0.682 ± 0.210	0.509 ± 0.128	0.595 ± 0.058	0.459 ± 0.054	$\mathbf{-0.05 \pm 0.114}$	$\mathbf{0.037 \pm 0.053}$
5	0.590 ± 0.057	0.417 ± 0.041	0.616 ± 0.031	0.486 ± 0.03	$\mathbf{0.069 \pm 0.076}$	$\mathbf{0.043 \pm 0.024}$

Table 11: For $\sigma = 1.5$, SL improves robustness of models to label shift for `Imbalanced-CIFAR10`. However we see a significantly greater robustness-accuracy tradeoff than with $\sigma = 0.1$ in Table 6, with all shifts resulting in performance drops.