ADAPTING TO LABEL SHIFT WITH BIAS-CORRECTED CALIBRATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Label shift refers to the phenomenon where the marginal probability $p(y)$ of observing a particular class changes between the training and test distributions, while the conditional probability $p(\boldsymbol{x}|y)$ stays fixed. This is relevant in settings such as medical diagnosis, where a classifier trained to predict disease based on observed symptoms may need to be adapted to a different distribution where the baseline frequency of the disease is higher. Given estimates of $p(y|\boldsymbol{x})$ from a predictive model, one can apply domain adaptation procedures including Expectation Maximization (EM) and Black-Box Shift Estimation (BBSE) to efficiently correct for the difference in class proportions between the training and test distributions. Unfortunately, modern neural networks typically fail to produce well-calibrated estimates of $p(y|\boldsymbol{x})$, reducing the effectiveness of these approaches. In recent years, Temperature Scaling has emerged as an efficient approach to combat miscalibration. However, the effectiveness of Temperature Scaling in the context of adaptation to label shift has not been explored. In this work, we study the impact of various calibration approaches on shift estimates produced by EM or BBSE. In experiments with image classification and diabetic retinopathy detection, we find that calibration consistently tends to improve shift estimation. In particular, calibration approaches that include class-specific bias parameters are significantly better than approaches that lack class-specific bias parameters, suggesting that reducing systematic bias in the calibrated probabilities is especially important for domain adaptation. Colab notebooks reproducing the results are available at (anonymized link): `https://github.com/blindauth/labelshiftexperiments`

1 INTRODUCTION

Imagine we train a classifier in country A to predict whether or not a person has a disease based on observed symptoms, and that we hope to deploy this classifier in country B, which has poorer access to healthcare. If the prevalence of the disease in country B is higher in than in country A, the classifier might systematically misdiagnose people as not having the disease. How can we adapt the classifier to cope with the difference in the baseline prevalence of the disease in the two countries?

Formally, let y denote our labels (e.g. whether or not a person is diseased), and let $\boldsymbol{x}$ denote the observed symptoms. Let us denote the joint distribution $(\boldsymbol{x}, y)$ in country A (our "source" domain) as $\mathbb{P}$, and let us denote the distribution in country B (our "target" domain, where we do not have labels) as $\mathbb{Q}$. How can we adapt a classifier trained to estimate $p(y|\boldsymbol{x})$ (the conditional probability in distribution $\mathbb{P}$) so that it can instead estimate $q(y|\boldsymbol{x})$ (the conditional probability in distribution $\mathbb{Q}$)? Absent assumptions about the nature of the shift between $\mathbb{P}$ and $\mathbb{Q}$, this problem is intractable. However, if the disease generates similar symptoms in both countries, we can assume that $p(\boldsymbol{x}|y) = q(\boldsymbol{x}|y)$, and that the shift in the joint distribution $q(\boldsymbol{x}, y)$ is due to a shift in the label proportion $q(y)$. Formally, we assume that $q(\boldsymbol{x}, y) = p(\boldsymbol{x}|y)q(y)$. This is known as *label shift* or *prior probability shift* (Amos, 2008), and it corresponds to anti-causal learning (i.e. predicting the cause y from its effects $\boldsymbol{x}$) (Schoelkopf et al., 2012). Anti-causal learning is appropriate for diagnosing diseases given observations of symptoms because diseases cause symptoms.

Given estimates of $p(y)$ and $p(y|\boldsymbol{x})$, there exist algorithms that can be applied to estimate $q(y)$ without needing to estimate $p(\boldsymbol{x}|y)$ (Saerens et al., 2002; Lipton et al., 2018). However, estimates of $p(y|\boldsymbol{x})$ derived from modern neural networks are often poorly calibrated (Guo et al., 2017), which

could in turn impact the estimates of $q(y)$. To reduce miscalibration, Guo et al. (2017) proposed Temperature Scaling, in which the logits of the output softmax are scaled by a temperature parameter to minimize the negative log likelihood (NLL) on a validation set. Guo et al. (2017) showed that Temperature Scaling is effective at minimizing a quantity they define as the Expected Calibration Error. However, the performance of Temperature Scaling (TS) in the context of domain adaptation to label shift has not, to our knowledge, been evaluated. We seek to bridge this gap. Our contributions are as follows:

- We explore the impact of calibration on two methods designed to perform domain adaptation to label shift: the EM algorithm of Saerens et al. (2002), and the Black Box Shift Estimation (BBSE) algorithm of Lipton et al. (2018). In both cases, we find that calibration significantly improves the quality of the label shift adaptation.
- We observed that TS alone can sometimes leave large systematic biases in the calibrated probabilities (**Fig.** 1), consistent with the findings of Kumar et al. (2019). Inspired by this observation, we tested whether variants of TS that contain class-specific bias parameters could correct for this systematic bias. We find this is indeed the case, and that such variants give superior performance on domain adapation to label shift - particularly for EM.
- We identify a theoretically-grounded strategy for computing the source-domain priors in EM-based domain adaptation that can be critically important when calibrated probability estimates have systematic bias. We find that when the source priors are computed in this way, label shift estimates computed though EM can perform surprisingly well compared to BBSE, even when predictions are not calibrated.
- We show that the calibration method that produces the largest improvement in Expected Calibration Error does not necessarily produce the largest improvement in negative log-likelihood (NLL), and that the NLL is more indicative of the improvement that a particular calibration approach gives in the context of domain adaptation to label shift.

2 Background

2.1 Temperature Scaling, Vector Scaling and Expected Calibration Error

Calibration has a long history in the machine learning literature (DeGroot and Fienberg, 1983; Platt, 1999; Zadrozny and Elkan; 2002; Niculescu-Mizil and Caruana, 2005; Kuleshov and Liang, 2015; Naeini et al., 2015; Kuleshov and Ermon, 2016). In the context of modern neural networks, Guo et al. (2017) showed that Temperature Scaling, a single-parameter variant of Platt Scaling (Platt, 1999), was effective at reducing miscalibration. Temperature scaling performs calibration by introducing a temperature parameter T to the logit vector of the softmax. Let $z(\boldsymbol{x}^k)$ represent a vector of the original softmax logits for example $\boldsymbol{x}^k$, and let y_i be a random variable representing the label for class i. With temperature scaling, we have $p(y_i|\boldsymbol{x}^k) = \frac{e^{z(\boldsymbol{x}^k)_i/T}}{\sum_j e^{z(\boldsymbol{x}^k)_j/T}}$, where T is optimized with respect to the Negative Log Likelihood (NLL) on a held-out portion of the training set, such as the validation set. Guo et al. (2017) compared TS to an approach defined as Vector Scaling (VS), where a different scaling parameter was used for each class along with class-specfic bias parameters. Formally, in vector scaling, $p(y_i|\boldsymbol{x}^k) = \frac{e^{(z(\boldsymbol{x}^k)_i W_i)+b_i}}{\sum_j e^{(z(\boldsymbol{x}^k)_j W_j)+b_j}}$. Guo et al. (2017) found that vector scaling had a tendency to perform

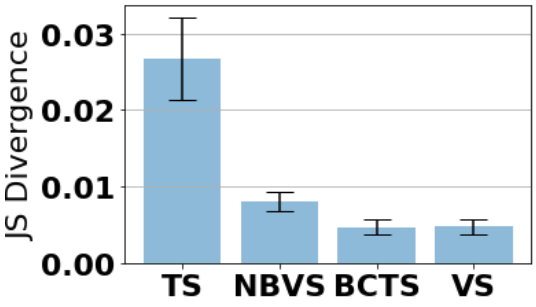

Figure 1: **Temperature Scaling exhibits systematic bias**. On CIFAR10 data, systematic bias was quantified by the Jensen-Shannon divergence between the true class label proportions and the average class predictions on a held-out test set drawn from the same distribution as the dataset used for calibration. TS: Temperature Scaling, NBVS: No-Bias Vector Scaling, BCTS: Bias-Corrected Temperature Scaling, VS: Vector Scaling. BCTS and VS had significantly lower systematic bias compared to TS and NBVS. Results are averaged over multiple models and dataset samples (**Sec. 4.1**).

slightly worse than TS as measured by a metric known as the Expected Calibration Error (Naeini et al., 2015). To compute the ECE, the predicted probabilities for the output class are partitioned into M equally spaced bins, and the weighted average of the difference between the bin's accuracy and the bin's confidence is computed, where the weights are determined by the proportion of examples falling in the bin. Formally, ECE $= \sum_{m=1}^{M} \frac{|B_m|}{n} |\text{acc}(B_m) - \text{conf}(B_m)|$, where n is the number of samples.

2.2 Label Shift Adaptation via Expectation Maximization

In a seminal paper on label shift adaptation, Saerens et al. (2002) proposed an EM algorithm for estimating the shift in the class priors between the training and test distributions. Let $\hat{q}^{(s)}(y = i)$ denote the estimate (from EM iteration s) of the prior probability $q(y = i)$ of observing class i in the test set. The algorithm proceeds as follows: first, $\hat{q}^{(0)}(y = i)$ is initialized to be equal to the class priors $\hat{p}(y = i)$ estimated from the training set. Then, the conditional probabilities in the E-step are computed as $\hat{q}^{(s)}(y = i|\boldsymbol{x}_k) = \frac{\frac{\hat{q}^{(s)}(y=i)}{\hat{p}(y=i)}\hat{p}(y=i|\boldsymbol{x}_k)}{\sum_{j=1}^{n}\frac{\hat{q}^{(s)}(y=j)}{\hat{p}(y=j)}\hat{p}(y=j|\boldsymbol{x}_k)}$. Finally, the prior estimates in the M-step are updated as $\hat{q}^{(s+1)}(y = i) = \frac{1}{N}\sum_{k=1}^{N}\hat{q}^{(s)}(y = i|\boldsymbol{x}_k)$, where N is the number of examples in the testing set. The E and M steps are iterated until convergence. As there is no need to estimate $p(\boldsymbol{x}|y)$ in any step of the EM procedure, the algorithm can scale to high-dimensional datasets. Note this procedure assumes the conditional probability estimates $\hat{p}(y = i|\boldsymbol{x}_k)$ are calibrated.

2.3 Label Shift Adaptation via Black Box Shift Estimation

Following the EM approach of Saerens et al. (2002), several additional approaches for labels shift adaptation have emerged (Chan and Ng; Storkey; Schoelkopf et al., 2012; Zhang et al., 2013; Lipton et al., 2018; Azizzadenesheli et al., 2019). Several of these approaches build estimates $p(x|y)$, which can scale poorly with dataset sizes and underperform on high-dimensional data (Lipton et al., 2018). Lipton et al. (2018) proposed Black-Box Shift Estimation (BBSE), which strives to efficiently estimate the weights $[\boldsymbol{w}]_i = \frac{q(y=i)}{p(y=i)}$ even in cases where the prediction model $\hat{p}(y = i|\boldsymbol{x}_k)$ is poorly calibrated or biased. BBSE proceeds as follows: let f be a function that accepts an input and returns the model's predicted class, $\boldsymbol{x}_k$ denote an example from a held-out portion of the training set, and $\boldsymbol{x}'_k$ denote an example from the testing set. The empirical estimate of $\boldsymbol{w}$, denoted as $\hat{\boldsymbol{w}}$, is computed as $\hat{\boldsymbol{w}} = \hat{\boldsymbol{C}}_{\hat{y},y}^{-1}\hat{\boldsymbol{u}}_{\hat{y}}$, where $[\hat{\boldsymbol{u}}_{\hat{y}}]_i = \frac{\sum_k \mathbb{1}\{f(\boldsymbol{x}'_k)=i\}}{m}$ and $[\hat{\boldsymbol{C}}_{\hat{y},y}]_{ij} = \frac{1}{n}\sum_k \mathbb{1}\{f(\boldsymbol{x}_k) = i \text{ and } y_k = j\}$. Because the approach above is not guaranteed to produce positive values for all elements of $\hat{\boldsymbol{w}}$, any negative elements of $\hat{\boldsymbol{w}}$ are set to 0 after they are estimated. Domain adaptation is then performed by retraining the model on the entire training set distribution with examples upweighted in accordance with $\hat{\boldsymbol{w}}$. Lipton et al. (2018) denote the version of BBSE described above as **BBSE-hard**. They also compare to a variant that they call **BBSE-soft**, which they describe as the case where where f outputs probabilities rather than hard classes. We interpreted this to mean $[\hat{\boldsymbol{u}}_{\hat{y}}]_i = \frac{\sum_k f(\boldsymbol{x}'_k)_i}{m}$ and $[\hat{\boldsymbol{C}}_{\hat{y},y}]_{ij} = \frac{1}{n}\sum_k f(\boldsymbol{x}_k)_i \mathbb{1}\{y_k = j\}$. Although BBSE is designed to work even with classifiers that are poorly calibrated or biased, in our experiments we found that BBSE-soft combined with an appropriate calibration method tended to outperform both the original BBSE-soft and BBSE-hard, neither of which used calibration. Note that BBSE requires a portion of the training set to be held out during the initial training phase in order to accurately estimate the confusion matrix $\hat{\boldsymbol{C}}_{\hat{\boldsymbol{y}},\boldsymbol{y}}$; in our experiments involving calibration, we use this same heldout set to calibrate the model.

3 Methods

3.1 No-Bias Vector Scaling and Bias-Corrected Temperature Scaling

As shown in **Fig. 1**, we often found that TS alone resulted in systematically biased estimates of $p(y_i|\boldsymbol{x}^k)$, while VS, a generalization of TS that contains both class-specific bias terms and class-specific scaling terms, did not exhibit as much systematic bias. Intrigued by this observation, we investigated the performance of two intermediaries between Temperature Scaling and Vector Scaling. The first, which we refer to as No Bias Vector Scaling (NBVS), is equivalent to vector scaling but with

all the class-specific bias parameters fixed at zero. The second, which we refer to as Bias-Corrected Temperature Scaling, is equivalent TS Scaling but with the addition of the class-specific bias terms from VS. As with TS and VS, the parameters are optimized to minimize the NLL on the validation set. Note that in the case of two-class (binary) classification, the parameterization of BCTS reduces to Platt Scaling (Platt, 1999). Thus, BCTS can be viewed as a multi-class generalization of Platt scaling.

3.2 DEFINING SOURCE-DOMAIN PRIORS IN THE EM ALGORITHM

The EM algorithm of (Saerens et al., 2002) requires the user to provide estimates of the source-domain prior class probabilities $\hat{p}(y = i)$. Let us consider two possible approaches to estimating these probabilities. The first, and most obvious, is to set $\hat{p}(y = i)$ to the expected value of the binary label $y = i$ over the source domain dataset. A second, slightly less obvious approach is to set it to the expected value of $\hat{p}(y = i|x)$ over the source domain dataset, formally denoted as $\mathbf{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}[\hat{p}(y = i|\boldsymbol{x})]$. If $\hat{p}(y = i|x)$ were unbiased, we would anticipate that the two approaches agree. However, depending on the calibration of $\hat{p}(y = i|\boldsymbol{x})$, this may not be the case, bringing us to:

Lemma 1: In the absence of domain shift and in the limit of sufficient data, the EM algorithm will converge to the original priors $\hat{p}(y = i)$ if and only if $\hat{p}(y = i) = \mathbf{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}[\hat{p}(y = i|\boldsymbol{x})]$.

Proof: Note that the EM algorithm will converge when $\hat{q}^{(s+1)}(y = i) = \hat{q}^{(s)}(y = i)$. From the M-step, we know that $\hat{q}^{(s+1)}(y = i) = \frac{1}{N}\sum_{k=1}^{N}\hat{q}^{(s)}(y = i|\boldsymbol{x}_k)$, where the examples $\boldsymbol{x}_k$ are drawn from the target distribution. Substituting the formula for $\hat{q}^{(s)}(y = i|\boldsymbol{x}_k)$ from the E-step, we have $\hat{q}^{(s+1)}(y = i) = \frac{1}{N}\sum_{k=1}^{N}\frac{\frac{\hat{q}^{(s)}(y=i)}{\hat{p}(y=i)}\hat{p}(y=i|\boldsymbol{x}_k)}{\sum_{j=1}^{n}\frac{\hat{q}^{(s)}(y=j)}{\hat{p}(y=j)}\hat{p}(y=j|\boldsymbol{x}_k)}$. To prove our lemma, we consider the scenario where $\hat{q}(y = i) = \hat{p}(y = i)$ and check whether convergence is attained. If the samples in the target distribution are drawn from the same distribution as the source, then in the limit of sufficient N, the value of $\hat{q}^{(s+1)}(y = 1)$ will approach $\mathbf{E}_{x\sim p(x)}\frac{\frac{1}{1}\hat{p}(y=i|\boldsymbol{x}_k)}{\sum_{j=1}^{n}\frac{1}{1}\hat{p}(y=j|\boldsymbol{x}_k)} = \mathbf{E}_{x\sim p(x)}\hat{p}(y = i|\boldsymbol{x}_k)$. Thus, convergence at $\hat{p}(y = i)$ will be attained if and only if $\hat{p}(y = i) = \mathbf{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}[\hat{p}(y = i|\boldsymbol{x})]$ ■

We reason that, in the absence of domain shift, it is desirable that EM converge to the original priors $\hat{p}(y = i)$. In light of Lemma 1, we set $\hat{p}(y = i)$ to be the average value of $\hat{p}(y = i|\boldsymbol{x})$ over the source-domain validation set (we use the validation set to avoid the effects of overfitting on the training set; this is the same validation set used for calibration). If we instead compute $\hat{p}(y = i)$ as the average of the binary label in the validation set, we observe very poor (even detrimental) performance with EM when the calibrated probabilities do not have bias correction (Tab. A.1).

If the source domain priors $\hat{p}(y = i)$ are defined to be the average of $\hat{p}(y = i|\boldsymbol{x})$ over the source domain samples (as outlined here), and $\hat{q}(y = i)$ is estimated in accordance with the standard EM update rules, we observe that the ratio of priors $\hat{q}(y = i)/\hat{p}(y = i)$ can serve as a surprisingly good estimate of the true ratio $q(y = i)/p(y = i)$, even when $\hat{p}(y = i)$ is systematically biased relative $p(y = i)$. See **Sec. 4.4** for more details.

3.3 METRICS FOR EVALUATING ADAPTATION TO LABEL SHIFT

3.3.1 JENSEN-SHANNON DIVERGENCE

The first metric we will consider in evaluating adaptation to label shift is the Jensen-Shannon divergence between the true target-domain priors and the target-domain priors that are estimated by a given domain adaptation method. Let us denote the true target-domain prior as $q(y = i)$ and the estimate as $q'(y = i)$. In the case of BBSE, $q'(y = i)$ can be calculated using the class weights $\hat{\boldsymbol{w}}_i$, which are intended to estimate $q(y = i)/p(y = i)$. Specifically, we use the formula $q'(y = i) = \frac{\hat{\boldsymbol{w}}_i p'(y=i)}{\sum_j \hat{\boldsymbol{w}}_j p'(y=j)}$, where $p'(y = i)$ is defined as the average of the true class labels in the source domain.

In the case of EM, some nuance is required. In **Sec. 3.2**, we noted that when performing EM, the source domain priors $\hat{p}(y = i)$ should be defined as the average of the predictions $\hat{p}(y = i|\boldsymbol{x})$ over the examples from the source domain (rather than being defined as the average of the labels). Once the EM algorithm has reached convergence, it will output an estimate $\hat{q}(y = i)$ of the target domain priors. Naively, it would seem that we could use $\hat{q}(y = i)$ as our estimate of $q(y = i)$ when

computing the Jensen-Shannon divergence. However, if $\hat{p}(y = i)$ is systematically biased, the target domain priors output by EM would likely also be systematically biased. We found that we can obtain superior estimates of $q(y = i)$ by leveraging the true labels in the source domain as follows: first, we average the labels in the source domain to obtain to obtain $p'(y = i)$, which is an unbiased estimate of $p(y = i)$. Then, we use the ratio $\hat{q}(y = i)/\hat{p}(y = i)$ from EM as a shift estimate. That is, we compute our estimate of the target domain priors using the formula $q'(y = i) = \frac{\frac{\hat{q}(y=i)}{\hat{p}(y=i)} p'(y=i)}{\sum_j \frac{\hat{q}(y=j)}{\hat{p}(y=j)} p'(y=j)}$. We find that target domain priors computed in this way can perform surprisingly well even when the predictions themselves are poorly calibrated (**Sec. 4.4**).

3.3.2 ACCURACY

The second metric we consider is the improvement in accuracy of the domain-adapted model predictions relative to using the original model predictions. Given the ratio $\hat{q}(y = i)/\hat{p}(y = i)$, the adapted model predictions can be computed as $\hat{q}(y = i|\boldsymbol{x}_k) = \frac{\frac{\hat{q}(y=i)}{\hat{p}(y=i)} \hat{p}(y=i|\boldsymbol{x}_k)}{\sum_j \frac{\hat{q}(y=j)}{\hat{p}(y=j)} \hat{p}(y=j|\boldsymbol{x}_k)}$, similar to the E-step of EM. For EM, we use these adapted predictions to assess accuracy. In the case of BBSE, Lipton et al. (2018) recommend retraining the model to obtain adapted predictions. Due to computational constraints, we did not perform model retraining, and thus we limit the comparisons of domain-adapted accuracy only to those calibration techniques that were used in conjunction with EM.

4 RESULTS

4.1 EXPERIMENTAL SETUP

We evaluated the efficacy of BBSE and EM coupled to different calibration approaches on CIFAR10, CIFAR100, and a diabetic retinopathy detection dataset. For our experiments on CIFAR10 and CIFAR100, we trained ten different models, each with a different random seed, using the code from Geifman and El-Yaniv (2017). For both CIFAR10 and CIFAR100, 10K examples of the training set were reserved as a held-out validation set. Dirichlet shift was simulated on the testing set by sampling with replacement in accordance with class proportions generated by a dirichlet distribution with uniform α values of 0.1 or 1.0 (smaller values of α result in more extreme label shift). Samples from the validation set were used for calibration, EM initialization and BBSE confusion matrix estimation. Accuracy and JS Divergence were reported on the label-shifted testing set, while the calibration metrics of NLL and ECE (with 15 bins) were reported on the unshifted testing set. In addition to exploring different degrees of dirichlet shift, we also investigated how the algorithms behaved when the number of samples used in the validation and testing set were varied. For example, in experiments with $n = 8000$, only 8000 samples from the validation set and 8000 samples from the shifted testing set were presented to the domain adaptation and calibration algorithms. For each model, for a given α and n, 10 trials were performed, where each trial consisted of a different sampling (without replacement) of the validation set as well as a different sampling of the dirichlet prior and the label-shifted testing set. This resulted in a total of 100 experiments (10 for each of the 10 different models). Statistical significance was calculated using a signed Wilcoxon test with a one-sided p-value threshold of 0.01. For CIFAR10, we also explored "tweak one" shift (Lipton et al., 2018), where the prior of the "cat" class was set to a parameter ρ and the remaining class priors were set to $(1 - \rho)/9$. We explored $\rho = 0.01$ and $\rho = 0.9$.

The Kaggle Diabetic Retinopathy dataset Kaggle (2015) is a collection of retinal fundus images and an associated "grade" from 0-4, where 0 indicates healthy and 1-4 indicate progressively more severe stages of retinopathy. For our experiments, we used the publicly-available pretrained model from De Fauw (2015), but it modified so as to make predictions on only one eye at a time (specifically, we supplied the mirror image of a given eye as the input for the second eye). Because test-set labels are unavailable, we separated the validation set used during the training of the model (consisting of 3514 examples) into "pseudo-validation" and "pseudo-test" sets. Specifically, for each of 100 trials, we sampled n examples from the original validation set without replacement to form a pseudo-validation set, and kept the remaining examples as the pseudo-test set. Calibration was performed on the pseudo-validation set, and calibration metrics of NLL and ECE were reported on the pseudo-test set. Domain shift was then simulated by sampling from the pseudo-test set in such a way that the proportion of "healthy" labels was set to a fraction ρ, and the relative proportions of diseased labels

was kept the same as in the source distribution. In the source distribution, $\rho = 0.73$; for the simulated domain shift, we explored $\rho = 0.5$ and $\rho = 0.9$.

4.2 CALIBRATION IMPROVES LABEL SHIFT ADAPTATION

Across datasets, we observe that calibration tends to improve label shift adaptation for both EM and BBSE (**Tables 1, 3, 4, 5, D.1, & E.1**). In particular, we observe that the variants of TS that contain bias-correction parameters (namely BCTS and VS) tend to be among the best-perfoming methods, particularly in the case of EM.

Shift Estimator	Calibration Method	$\alpha = 0.1$			$\alpha = 1.0$		
		n=2000	n=4000	n=8000	n=2000	n=4000	n=8000
EM	None	6.986; 2.77	6.926; 3.17	6.938; 3.31	1.968; 3.36	2.016; 3.44	2.055; 3.69
EM	TS	7.251; 1.68	7.2; 2.13	7.217; 2.21	2.127; 2.83	2.172; 2.92	2.204; 3.05
EM	NBVS	**7.324; 1.63**	**7.314; 1.59**	7.314; 1.69	2.5; 1.46	2.592; 1.47	2.631; 1.45
EM	BCTS	**7.328; 1.69**	**7.337; 1.42**	**7.347; 1.4**	**2.593; 0.98**	**2.664; 1.0**	2.688; 1.09
EM	VS	7.255; 2.23	**7.331; 1.69**	**7.372; 1.39**	**2.548; 1.37**	**2.652; 1.17**	**2.724; 0.72**

Table 1: **CIFAR10: Comparison of calibration methods when using EM adaptation to dirchlet shift, with Δ%accuracy as the metric.** Value before the semicolon is the average change in %accuracy relative to a baseline of no adaptation. Value after the semicolon is the average rank compared to other methods in the same column. α represents the dirichlet shift parameter, n represents the sample size for both the validation set and the label-shifted test set. Bold values in a column are not significantly different from the best performing method in the column, as measured by a paired Wilcoxon test at $p \leq 0.01$. See **Sec. 4.1** for details on the experimental setup.

Shift Estimator	Calibration Method	$\rho = 0.01$			$\rho = 0.9$		
		n=2000	n=4000	n=8000	n=2000	n=4000	n=8000
EM	None	0.784; 2.91	0.798; 3.04	0.761; 3.31	16.304; 3.79	16.356; 3.84	16.369; 3.92
EM	TS	0.807; 2.92	0.807; 3.14	0.775; 3.4	17.193; 2.48	17.26; 2.67	17.288; 2.75
EM	NBVS	**1.149; 1.31**	1.172; 1.56	1.199; 1.39	17.588; 1.52	17.674; 1.51	17.738; 1.68
EM	BCTS	**1.175; 1.38**	1.224; 1.27	1.262; 1.15	**17.724; 1.09**	17.779; 1.17	17.84; 1.24
EM	VS	**1.182; 1.48**	**1.258; 0.99**	**1.301; 0.75**	**17.727; 1.12**	**17.874; 0.81**	**17.988; 0.41**

Table 2: **CIFAR10: Comparison of calibration methods when using EM adaptation to "tweak-one" shift, with Δ%accuracy as the metric.** Analogous to **Table 1**

Shift Estimator	Calibration Method	$\alpha = 0.1$			$\alpha = 1.0$		
		n=2000	n=4000	n=8000	n=2000	n=4000	n=8000
BBSE-soft	None	0.068; 2.47	0.056; 2.43	0.049; 2.46	0.027; 2.48	0.019; 2.29	0.014; 2.2
BBSE-soft	TS	**0.067; 1.88**	**0.056; 2.0**	0.048; 1.98	**0.026; 2.0**	**0.019; 1.97**	0.013; 1.88
BBSE-soft	NBVS	0.067; 2.18	**0.055; 1.96**	0.048; 2.13	**0.026; 1.98**	**0.018; 1.79**	0.013; 1.89
BBSE-soft	BCTS	**0.066; 1.71**	**0.055; 1.73**	**0.047; 1.7**	**0.025; 1.77**	**0.018; 1.91**	0.014; 1.97
BBSE-soft	VS	**0.066; 1.76**	**0.055; 1.88**	**0.047; 1.73**	**0.025; 1.77**	**0.018; 2.04**	0.014; 2.06

Table 3: **CIFAR10: Comparison of calibration methods when using BBSE adaptation to dirichlet shift, with JS Divergence (Sec. 3.3) as the metric.** Analogous to **Table 1**.

Shift Estimator	Calibration Method	$\alpha = 0.1$			$\alpha = 1.0$		
		n=7000	n=8500	n=10000	n=7000	n=8500	n=10000
EM	None	14.41; 4.0	14.483; 4.0	14.463; 4.0	12.25; 4.0	12.292; 4.0	12.319; 4.0
EM	TS	26.112; 1.63	26.101; 1.64	26.048; 1.68	21.625; 1.82	21.638; 1.9	21.622; 1.9
EM	NBVS	26.332; 1.6	26.323; 1.73	26.464; 1.7	21.588; 1.86	21.711; 1.91	21.708; 2.04
EM	BCTS	26.485; 1.67	26.638; 1.47	**26.731; 1.44**	**21.907; 1.17**	**22.004; 1.23**	22.015; 1.24
EM	VS	**26.889; 1.1**	**26.901; 1.16**	**26.954; 1.18**	**21.94; 1.15**	**22.097; 0.96**	**22.183; 0.82**

Table 4: **CIFAR100: Comparison of calibration methods when using EM adaptation to dirichlet shift, with Δ%accuracy as the metric.** Analogous to **Table 1**.

4.3 BIAS-CORRECTION IMPROVES NLL OF CALIBRATED PREDICTIONS

We find that bias-corrected versions of TS (namely BCTS and VS) tend to yield the best NLL on an unshifted test set, even if they do not always yield the best ECE (**Tables 6, 7 & E.2**). Recall

Shift Estimator	Calibration Method	$\rho = 0.5$			$\rho = 0.9$		
		n=500	n=1000	n=1500	n=500	n=1000	n=1500
EM	None	1.926; 3.09	2.076; 3.49	2.196; 3.64	1.296; 3.42	1.375; 3.81	1.477; 3.8
EM	TS	1.902; 2.96	2.225; 3.17	2.495; 3.13	1.626; 3.01	1.923; 2.88	1.973; 2.97
EM	NBVS	3.23; 1.69	3.789; 1.49	4.062; 1.54	2.074; 2.44	2.266; 2.24	2.405; 2.17
EM	BCTS	**3.766; 0.88**	**4.356; 0.74**	**4.58; 0.82**	**3.548; 0.35**	**3.567; 0.36**	**3.722; 0.44**
EM	VS	**3.67; 1.38**	**4.278; 1.11**	**4.545; 0.87**	**3.5; 0.78**	**3.57; 0.71**	**3.746; 0.62**

Table 5: **Kaggle Diabetic Retinopathy: Comparison of calibration methods when using EM adaptation to domain shift, with Δ%accuracy as the metric.** ρ represents proportion of healthy examples in shifted domain; source distribution has $\rho = 0.73$. Analogous to **Table 1**, but with a different type of domain shift (described in **Sec. 4.1**).

that the calibration metrics are optimized with respect to NLL on the validation set. Empirically, we find the NLL corresponds better with the improvement that a calibration method will give to domain adaptation (see **Sec. B**).

Calibration Method	NLL			ECE		
	n=2000	n=4000	n=8000	n=2000	n=4000	n=8000
None	0.299; 4.0	0.299; 4.0	0.299; 4.0	2.726; 4.0	2.726; 4.0	2.726; 4.0
TS	0.291; 2.99	0.291; 3.0	0.291; 3.0	**1.069; 1.23**	1.06; 1.83	1.027; 2.12
NBVS	0.277; 1.67	0.275; 1.92	0.274; 1.99	1.109; 1.51	1.023; 1.51	**0.952; 1.35**
BCTS	**0.274; 0.34**	**0.272; 0.54**	0.271; 0.71	**1.06; 1.02**	**0.987; 1.02**	**0.937; 1.06**
VS	0.275; 1.0	**0.272; 0.54**	**0.271; 0.3**	1.161; 2.24	1.035; 1.64	0.976; 1.47

Table 6: **CIFAR10: NLL and ECE for different calibration methods**. Metrics were computed on a test set that had the same distribution as the validation set. Value before the semicolon is the average of the metric over all the runs. Value after the semicolon is the average rank of the method relative to other methods in the column. Bold values in a column are not significantly different from the best performing method in the column, as measured by a paired Wilcoxon test at $p \leq 0.01$. See **Sec. 4.1** for details on the experimental setup.

Calibration Method	NLL			ECE		
	n=7000	n=8500	n=10000	n=7000	n=8500	n=10000
None	1.735; 4.0	1.735; 4.0	1.735; 4.0	20.041; 4.0	20.041; 4.0	20.041; 4.0
TS	1.286; 3.0	1.286; 3.0	1.286; 3.0	3.134; 2.87	3.151; 2.87	3.135; 2.9
NBVS	1.241; 2.0	1.24; 2.0	1.239; 2.0	**2.263; 0.09**	**2.281; 0.1**	**2.324; 0.1**
BCTS	1.234; 0.71	1.233; 0.9	1.232; 1.0	2.879; 2.11	2.9; 2.12	2.881; 2.1
VS	**1.234; 0.29**	**1.231; 0.1**	**1.229; 0.0**	2.458; 0.93	2.48; 0.91	2.456; 0.9

Table 7: **CIFAR100: NLL and ECE for different calibration methods**. Analogous to **Table 6**.

4.4 EM IS SURPRISINGLY EFFECTIVE AT ESTIMATING SHIFT RATIOS

In this work, we made the surprising observation that when the source domain priors are initialized to be the average of the predicted probabilities on the source samples (as discussed in Sec. 3.2), the ratio $\hat{q}(y = i)/\hat{p}(y = i)$ estimated by EM can be a surprisingly good estimate of the true ratio $q(y = i)/p(y = i)$ (**Tables 8, 9, C.2, E.3**). In fact, EM performs competitively with BBSE even when predicted probabilities retain systematic bias (as is the case with TS - see **Fig. 1**). BBSE shows advantages over EM in the presence of no calibration, although EM still sometimes performs unexpectedly well. However, EM performed in the absence of good calibration lacks the theoretical guarantees of BBSE, and therefore may not be preferable if the quality of the calibration is doubtful.

5 DISCUSSION

In this work, we explored the effect of calibration on procedures designed to perform domain adaptation to label shift. In experiments on CIFAR10, CIFAR100 and diabetic retinopathy detection, we found that calibration consistently improves the quality of the label shift adaptation for both EM

Shift Estimator	Calibration Method	$\rho = 0.1$			$\rho = 1.0$		
		n=2000	n=4000	n=8000	n=2000	n=4000	n=8000
EM	None	**0.035; 0.09**	**0.033; 0.11**	**0.033; 0.21**	**0.021; 0.37**	0.018; 0.54	0.016; 0.69
BBSE-soft	None	0.068; 0.91	0.056; 0.89	0.049; 0.79	0.027; 0.63	0.019; 0.46	**0.014; 0.31**
EM	TS	**0.024; 0.02**	**0.022; 0.04**	**0.021; 0.07**	**0.019; 0.29**	0.017; 0.47	0.014; 0.59
BBSE-soft	TS	0.067; 0.98	0.056; 0.96	0.048; 0.93	0.026; 0.71	0.019; 0.53	0.013; 0.41
EM	NBVS	**0.021; 0.01**	**0.017; 0.02**	**0.017; 0.02**	**0.017; 0.15**	**0.013; 0.19**	**0.01; 0.23**
BBSE-soft	NBVS	0.067; 0.99	0.055; 0.98	0.048; 0.98	0.026; 0.85	0.018; 0.81	0.013; 0.77
EM	BCTS	**0.02; 0.01**	**0.016; 0.01**	**0.015; 0.04**	**0.016; 0.11**	**0.012; 0.17**	**0.009; 0.18**
BBSE-soft	BCTS	0.066; 0.99	0.055; 0.99	0.047; 0.96	0.025; 0.89	0.018; 0.83	0.014; 0.82
EM	VS	**0.023; 0.01**	**0.017; 0.0**	**0.015; 0.03**	**0.016; 0.15**	**0.013; 0.14**	**0.009; 0.22**
BBSE-soft	VS	0.066; 0.99	0.055; 1.0	0.047; 0.97	0.025; 0.85	0.018; 0.86	0.014; 0.78

Table 8: **CIFAR10: Comparison of EM and BBSE (dirichlet shift)**. Value before the semicolon is the avg. JS divergence between true and estimated target priors (**Sec. 3.3**). Value after the semicolon is the avg. rank of a method relative to the other in the pair. A bold value is significantly better than the non-bold value in the pair (paired Wilcoxon test, $p \leq 0.01$). See **Sec. 4.1** for experimental setup.

Shift Estimator	Calibration Method	$\alpha = 0.1$			$\alpha = 1.0$		
		n=7000	n=8500	n=10000	n=7000	n=8500	n=10000
EM	None	**0.233; 0.34**	**0.232; 0.41**	0.232; 0.45	0.232; 0.84	0.231; 0.93	0.23; 0.97
BBSE-soft	None	0.248; 0.66	0.241; 0.59	0.237; 0.55	**0.21; 0.16**	**0.204; 0.07**	**0.198; 0.03**
EM	TS	**0.113; 0.0**	**0.113; 0.0**	**0.113; 0.0**	**0.109; 0.0**	**0.107; 0.0**	**0.106; 0.0**
BBSE-soft	TS	0.224; 1.0	0.218; 1.0	0.214; 1.0	0.187; 1.0	0.181; 1.0	0.176; 1.0
EM	NBVS	**0.119; 0.01**	**0.119; 0.01**	**0.119; 0.01**	**0.118; 0.0**	**0.117; 0.0**	**0.116; 0.0**
BBSE-soft	NBVS	0.226; 0.99	0.22; 0.99	0.216; 0.99	0.189; 1.0	0.183; 1.0	0.178; 1.0
EM	BCTS	**0.118; 0.01**	**0.117; 0.01**	**0.117; 0.01**	**0.112; 0.0**	**0.111; 0.0**	**0.11; 0.0**
BBSE-soft	BCTS	0.224; 0.99	0.218; 0.99	0.214; 0.99	0.187; 1.0	0.181; 1.0	0.176; 1.0
EM	VS	**0.113; 0.01**	**0.112; 0.0**	**0.112; 0.01**	**0.111; 0.0**	**0.108; 0.0**	**0.107; 0.0**
BBSE-soft	VS	0.226; 0.99	0.22; 1.0	0.215; 0.99	0.188; 1.0	0.182; 1.0	0.177; 1.0

Table 9: **CIFAR100: Comparison of EM and BBSE (dirichlet shift)**. Analogous to **Table 8**.

and BBSE, with EM benefiting particularly well when the calibration approach contains class-specific bias parameters that can reduce systematic bias in the class probabilities.

In addition, we observed that when the calibrated probabilities retain systematic bias, domain adaptation to EM is sensitive to the strategy used to compute the source-domain priors. If the source-domain priors $\hat{p}(y = i)$ are not defined in a way that mirrors the systematic bias in the predicted probabilities $\hat{p}(y = i|x)$, then EM will estimate a label shift even if the target domain is identical to the source domain (**Lemma 1**) and can produce highly detrimental results (**Tables A.1 & A.2**).

If, however, the source domain priors for EM are initialized as we recommend in **Sec. 3.2**, the shift ratios $\hat{q}(y = i)/\hat{p}(y = i)$ from EM can serve as surprisingly good estimates of the true shift estimate $q(y = i)/p(y = i)$ (**Sec. 4.4**), and occasionally outperform BBSE even when the probabilities are not well calibrated. However, EM performed in the absence of good calibration lacks the theoretical guarantees of BBSE, and thus might not be preferable in practice.

Finally, we observed that calibration strategies that included class-specific bias parameters, namely VS and BCTS, tend to produce superior NLL on the unshifted held-out test set, even when they do not produce the best ECE (**Sec. 4.3**). Further, the NLL appears to correspond better to the improvement in domain adaptation to label shift (**Sec. B**). This suggests ECE and NLL may offer complementary measures of calibration quality; the ECE is concerned only with the output for the class with the highest probability (i.e. the predicted class), whereas the NLL considers the probabilities output for all classes. In many applications, we are concerned only with the calibration quality of the predicted class; however, for domain adaptation, the predicted probabilities of all classes come into play. Calibrating these probabilities in class-specific ways works well in the face of label shifts.

REFERENCES

Storkey Amos. When training and test sets are different: Characterizing learning transfer. In *Dataset Shift in Machine Learning*. The MIT Press, 2008.

Kamyar Azizzadenesheli, Anqi Liu, Fanny Yang, and Animashree Anandkumar. Regularized learning for domain adaptation under label shifts. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=rJl0r3R9KX.

Yee Seng Chan and Hwee Tou Ng. Word sense disambiguation with distribution estimation.

Jeffrey De Fauw. Jeffreydf/kaggle_diabetic_retinopathy: Fifth place solution of the kaggle diabetic retinopathy competition. https://github.com/JeffreyDF/kaggle_diabetic_retinopathy, Oct 2015. (Accessed on 01/22/2019).

Morris H DeGroot and Stephen E Fienberg. The comparison and evaluation of forecasters. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 32(1-2):12–22, 1983.

Y. Geifman and R. El-Yaniv. Selective Classification for Deep Neural Networks. *ArXiv e-prints*, 2017.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. June 2017.

Kaggle. Kaggle competition on diabetic retinopathy detection. https://www.kaggle.com/c/diabetic-retinopathy-detection#description, 2015.

Volodymyr Kuleshov and Stefano Ermon. Reliable confidence estimation via online learning. *arXiv preprint arXiv:1607.03594*, pages 2586–2594, 2016.

Volodymyr Kuleshov and Percy S Liang. Calibrated structured prediction. In *Advances in Neural Information Processing Systems*, pages 3474–3482, 2015.

Ananya Kumar, Percy Liang, and Tengyu Ma. Verified uncertainty calibration, 2019.

Zachary Lipton, Yu-Xiang Wang, and Alexander Smola. Detecting and correcting for label shift with black box predictors. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 3122–3130, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL http://proceedings.mlr.press/v80/lipton18a.html.

Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using bayesian binning. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

Alexandru Niculescu-Mizil and Rich Caruana. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning*, pages 625–632. ACM, 2005.

John C Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *ADVANCES IN LARGE MARGIN CLASSIFIERS*, 1999.

Marco Saerens, Patrice Latinne, and Christine Decaestecker. Adjusting the outputs of a classifier to new a priori probabilities: a simple procedure. *Neural Comput.*, 14(1):21–41, January 2002.

Bernhard Schoelkopf, Dominik Janzing, Jonas Peters, Eleni Sgouritsa, Kun Zhang, and Joris Mooij. On causal and anticausal learning. June 2012.

Amos Storkey. When training and test sets are different: characterizing learning transfer.

Bianca Zadrozny and Charles Elkan. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. Citeseer.

Bianca Zadrozny and Charles Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 694–699. ACM, 2002.

Kun Zhang, Bernhard Schölkopf, Krikamol Muandet, and Zhikun Wang. Domain adaptation under target and conditional shift. In Sanjoy Dasgupta and David McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 819–827, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR. URL http://proceedings.mlr.press/v28/zhang13d.html.

A COMPARISON OF STRATEGIES FOR INITIALIZING EM SOURCE PROBABILITIES

Shift Estimator	Calibration Method	$\rho = 0.5$			$\rho = 0.9$		
		n=500	n=1000	n=1500	n=500	n=1000	n=1500
EM: source priors from preds	None	**1.926; 0.0**	**2.076; 0.0**	**2.196; 0.0**	**1.296; 0.26**	**1.375; 0.17**	**1.477; 0.14**
EM: source priors from labels	None	-3.488; 1.0	-3.541; 1.0	-3.382; 1.0	0.782; 0.74	0.937; 0.83	1.043; 0.86
EM: source priors from preds	TS	**1.902; 0.0**	**2.225; 0.0**	**2.495; 0.0**	**1.626; 0.0**	**1.923; 0.0**	**1.973; 0.0**
EM: source priors from labels	TS	-56.162; 1.0	-62.552; 1.0	-64.195; 1.0	-69.146; 1.0	-76.619; 1.0	-83.083; 1.0
EM: source priors from preds	NBVS	**3.23; 0.0**	**3.789; 0.0**	**4.062; 0.0**	**2.074; 0.02**	**2.266; 0.01**	**2.405; 0.02**
EM: source priors from labels	NBVS	-9.448; 1.0	-5.134; 1.0	-4.772; 1.0	-2.616; 0.98	0.431; 0.99	0.631; 0.98
EM: source priors from preds	BCTS	**3.766; 0.0**	4.356; 0.03	4.58; 0.01	3.548; 0.0	3.567; 0.01	3.722; 0.01
EM: source priors from labels	BCTS	3.764; 1.0	**4.357; 0.97**	4.58; 0.99	3.548; 1.0	**3.568; 0.99**	**3.723; 0.99**
EM: source priors from preds	VS	3.67; 0.08	4.278; 0.08	4.545; 0.08	3.5; 0.03	3.57; 0.03	3.746; 0.03
EM: source priors from labels	VS	3.662; 0.92	4.278; 0.92	**4.559; 0.92**	**3.506; 0.97**	**3.572; 0.97**	3.746; 0.97

Table A.1: **The strategy for computing EM source priors heavily affects domain adaptation if probabilities retain systematic bias**. Value before the semicolon is the average improvement in %accuracy (across 100 trials) caused by applying domain adaptation to the predictions on a diabetic retinopathy prediction task. Value after the semicolon is the average rank of a particular method relative to the other method in the pair. Domain shift is induced by varying the proportion of "healthy" examples ρ; in the source distribution, $\rho = 0.73$. We see that calibration methods that lack class-specific bias parameters (i.e. no calibration, TS and NBVS) can hurt domain adaptation if source priors are initialized by averaging true labels rather than the predicted probabilities. A bold value in a pair is significantly better than the non-bold value according to a paired Wilcoxon test at $p \leq 0.01$. See **Table A.2** for analogous results using JS Div. See **Sec. 4.1** for details on the experimental setup.

Shift Estimator	Calibration Method	$\rho = 0.5$			$\rho = 0.9$		
		n=500	n=1000	n=1500	n=500	n=1000	n=1500
EM: source priors from preds	None	**0.077; 0.0**	**0.059; 0.0**	**0.054; 0.0**	**0.111; 0.4**	**0.1; 0.37**	**0.102; 0.32**
EM: source priors from labels	None	0.253; 1.0	0.256; 1.0	0.256; 1.0	0.125; 0.6	0.116; 0.63	0.114; 0.68
EM: source priors from preds	TS	**0.09; 0.0**	**0.068; 0.0**	**0.061; 0.0**	**0.104; 0.0**	**0.094; 0.0**	**0.094; 0.0**
EM: source priors from labels	TS	0.61; 1.0	0.628; 1.0	0.647; 1.0	0.629; 1.0	0.639; 1.0	0.643; 1.0
EM: source priors from preds	NBVS	**0.107; 0.0**	**0.089; 0.0**	**0.079; 0.0**	**0.11; 0.0**	**0.1; 0.0**	**0.102; 0.0**
EM: source priors from labels	NBVS	0.348; 1.0	0.327; 1.0	0.32; 1.0	0.214; 1.0	0.192; 1.0	0.188; 1.0
EM: source priors from preds	BCTS	0.111; 0.4	0.095; 0.44	0.082; 0.41	0.078; 0.49	0.065; 0.54	0.063; 0.52
EM: source priors from labels	BCTS	0.111; 0.6	0.095; 0.56	0.082; 0.59	0.078; 0.51	0.065; 0.46	0.063; 0.48
EM: source priors from preds	VS	0.108; 0.43	0.088; 0.5	0.076; 0.48	0.077; 0.58	0.062; 0.45	0.059; 0.54
EM: source priors from labels	VS	0.108; 0.57	0.088; 0.5	0.076; 0.52	0.077; 0.42	0.062; 0.55	0.059; 0.46

Table A.2: Similar to **Table A.1**, but using Jensen-Shannon Divergence as a metric to assess the quality of domain adaptation. See **Sec. 3.3** for a description of how the Jensen-Shannon Divergence metric is calculated.

B NLL CORRESPONDS BETTER TO BENEFITS IN LABEL SHIFT ADAPTATION

To investigate whether NLL or ECE corresponded better to the benefits offered by a calibration method in the context of label shift adaptation, we adopted the following strategy: in a given experimental run, we identified the calibration method that provided the best NLL (or ECE) on the unshifted test set. We then looked at the performance of label shift adaptation using this calibration method. Note that the calibration method selected can differ from one run to the next. Across datasets, we observed that, by and large, selecting a calibration method according to the NLL produced better performance after domain adaptation as compared to selecting a calibration method according to ECE. Results are show in **Tables B.1, B.2, B.3, B.4, B.5, B.6, B.7 & B.8**.

Shift Estimator	Calibration Method	$\alpha = 0.1$			$\alpha = 1.0$		
		n=2000	n=4000	n=8000	n=2000	n=4000	n=8000
EM	Best NLL	7.332; 0.3	7.326; 0.32	**7.37; 0.28**	2.593; 0.36	**2.664; 0.09**	**2.688; 0.06**
EM	Best ECE	7.298; 0.7	7.302; 0.68	7.318; 0.72	2.548; 0.64	2.172; 0.91	2.204; 0.94

Table B.1: **CIFAR10: NLL vs. ECE, metric: Δ%accuracy, dirichlet shift**. Entry in "calibration method" column indicates how the calibration method for any given run was selected: either according to whether it produced the best NLL or whether it produced the best ECE, where NLL and ECE were calculated on the unshifted test set. Value before the semicolon is the average change in %accuracy relative to unadapted predictions. Value after the semicolon is the average rank of the given metric relative to the other metric in the pair. A bold value is significantly better than the non-bold value in the pair using a paired Wilcoxon test at $p \leq 0.01$. See **Sec. 4.1** for details on the experimental setup.

Shift Estimator	Calibration Method	$\rho = 0.01$			$\rho = 0.9$		
		n=2000	n=4000	n=8000	n=2000	n=4000	n=8000
EM	Best NLL	**1.192; 0.17**	**1.253; 0.21**	**1.301; 0.15**	17.724; 0.47	**17.779; 0.08**	**17.84; 0.07**
EM	Best ECE	1.053; 0.83	1.149; 0.79	1.16; 0.85	17.727; 0.53	17.26; 0.92	17.288; 0.93

Table B.2: **CIFAR10: NLL vs. ECE, metric: Δ%accuracy, "tweak-one" shift**. Analogous to **Table B.1**. The "tweak-one" shift strategy is explained in **Sec. 4.1**.

Shift Estimator	Calibration Method	$\alpha = 0.1$			$\alpha = 1.0$		
		n=2000	n=4000	n=8000	n=2000	n=4000	n=8000
EM	Best NLL	2.003; 0.33	1.625; 0.4	**1.488; 0.36**	1.6; 0.38	**1.24; 0.14**	**0.918; 0.12**
EM	Best ECE	2.144; 0.67	1.635; 0.6	1.642; 0.64	1.647; 0.62	1.714; 0.86	1.398; 0.88
BBSE-soft	Best NLL	6.636; 0.41	5.493; 0.35	4.689; 0.49	2.543; 0.49	1.804; 0.47	1.356; 0.55
BBSE-soft	Best ECE	6.65; 0.59	5.526; 0.65	4.703; 0.51	2.55; 0.51	1.87; 0.53	1.339; 0.45

Table B.3: **CIFAR10: NLL vs. ECE, metric: JS Divergence, dirichlet shift**. Analogous to **Table B.1**, but using JS Divergence (**Sec. 3.3**) as the metric rather than change in %accuracy.

Shift Estimator	Calibration Method	$\rho = 0.01$			$\rho = 0.9$		
		n=2000	n=4000	n=8000	n=2000	n=4000	n=8000
EM	Best NLL	**1.566; 0.29**	**1.0; 0.33**	**0.793; 0.29**	2.787; 0.39	**2.339; 0.13**	**2.132; 0.04**
EM	Best ECE	1.695; 0.71	1.099; 0.67	0.906; 0.71	2.908; 0.61	3.388; 0.87	3.223; 0.96
BBSE-soft	Best NLL	2.012; 0.4	1.409; 0.51	0.985; 0.55	7.431; 0.59	**5.737; 0.25**	**3.84; 0.16**
BBSE-soft	Best ECE	2.005; 0.6	**1.389; 0.49**	**0.959; 0.45**	7.393; 0.41	6.214; 0.75	4.328; 0.84

Table B.4: **CIFAR10: NLL vs. ECE, metric: JS Divergence, "tweak-one" shift**. Analogous to **Table B.1**.

Shift Estimator	Calibration Method	$\alpha = 0.1$			$\alpha = 1.0$		
		n=7000	n=8500	n=10000	n=7000	n=8500	n=10000
EM	Best NLL	**26.889; 0.3**	**26.901; 0.31**	**26.954; 0.31**	**21.958; 0.27**	**22.106; 0.26**	**22.183; 0.2**
EM	Best ECE	26.332; 0.7	26.323; 0.69	26.464; 0.69	21.63; 0.73	21.77; 0.74	21.777; 0.8

Table B.5: **CIFAR100: NLL vs. ECE, metric: Δ%accuracy, dirichlet shift**. Analogous to **Table B.1**.

Shift Estimator	Calibration Method	$\alpha = 0.1$			$\alpha = 1.0$		
		n=7000	n=8500	n=10000	n=7000	n=8500	n=10000
EM	Best NLL	**0.113; 0.31**	**0.112; 0.27**	**0.112; 0.29**	**0.11; 0.15**	**0.108; 0.1**	**0.107; 0.1**
EM	Best ECE	0.119; 0.69	0.119; 0.73	0.119; 0.71	0.117; 0.85	0.116; 0.9	0.115; 0.9
BBSE-soft	Best NLL	0.226; 0.41	0.22; 0.35	**0.215; 0.39**	**0.188; 0.21**	**0.182; 0.26**	**0.177; 0.23**
BBSE-soft	Best ECE	0.226; 0.59	0.22; 0.65	0.216; 0.61	0.189; 0.79	0.183; 0.74	0.178; 0.77

Table B.6: **CIFAR100: NLL vs. ECE, metric: JS Divergence, dirichlet shift**. Analogous to **Table B.1**.

Shift Estimator	Calibration Method	$\rho = 0.5$			$\rho = 0.9$		
		n=500	n=1000	n=1500	n=500	n=1000	n=1500
EM	Best NLL	**3.79; 0.21**	**4.315; 0.26**	**4.543; 0.19**	**3.548; 0.02**	3.57; 0.0	**3.746; 0.02**
EM	Best ECE	3.49; 0.79	4.099; 0.74	4.179; 0.81	2.074; 0.98	3.57; 1.0	2.405; 0.98
BBSE-soft	Best NLL	1.546; 0.22	**3.584; 0.26**	**4.164; 0.19**	**3.282; 0.07**	3.442; 0.0	**3.621; 0.06**
BBSE-soft	Best ECE	2.048; 0.78	3.026; 0.74	3.76; 0.81	1.93; 0.93	3.442; 1.0	2.759; 0.94

Table B.7: **Kaggle Diabetic Retinopathy: NLL vs. ECE, metric: Δ%accuracy, "change proportion of healthy" shift**. Analogous to **Table B.1**.

Shift Estimator	Calibration Method	$\rho = 0.5$			$\rho = 0.9$		
		n=500	n=1000	n=1500	n=500	n=1000	n=1500
EM	Best NLL	0.11; 0.42	0.093; 0.31	0.079; 0.33	**0.078; 0.08**	0.062; 0.0	**0.059; 0.07**
EM	Best ECE	0.104; 0.58	0.092; 0.69	0.079; 0.67	0.11; 0.92	0.062; 1.0	0.102; 0.93
BBSE-soft	Best NLL	0.166; 0.37	0.12; 0.32	**0.096; 0.31**	**0.107; 0.24**	0.079; 0.0	**0.077; 0.32**
BBSE-soft	Best ECE	0.158; 0.63	0.123; 0.68	0.101; 0.69	0.125; 0.76	0.079; 1.0	0.086; 0.68

Table B.8: **Kaggle Diabetic Retinopathy: NLL vs. ECE, metric: JS Divergence, "change proportion of healthy" shift**. Analogous to **Table B.1**.

C CIFAR10 SUPPLEMENTARY TABLES

Shift Estimator	Calibration Method	$\rho = 0.01$			$\rho = 0.9$		
		n=2000	n=4000	n=8000	n=2000	n=4000	n=8000
BBSE-soft	None	0.021; 2.37	0.014; 2.05	0.01; 2.16	0.08; 2.97	0.064; 3.15	0.045; 3.25
BBSE-soft	TS	**0.021; 2.07**	**0.014; 1.65**	**0.01; 1.9**	0.078; 2.31	0.062; 2.47	0.043; 2.76
BBSE-soft	NBVS	**0.02; 1.8**	**0.014; 2.08**	**0.01; 1.94**	**0.075; 1.75**	0.058; 1.74	0.04; 1.75
BBSE-soft	BCTS	**0.02; 1.8**	**0.014; 2.12**	**0.01; 1.8**	**0.074; 1.56**	0.057; 1.52	**0.038; 1.26**
BBSE-soft	VS	**0.02; 1.96**	**0.014; 2.1**	**0.01; 2.2**	**0.074; 1.41**	**0.057; 1.12**	**0.038; 0.98**

Table C.1: **CIFAR10: Comparison of calibration methods when using BBSE adaptation to "tweak-one" shift, with JS Divergence (Sec. 3.3) as the metric.** Analogous to **Table 1**

Shift Estimator	Calibration Method	$\rho = 0.01$			$\rho = 0.9$		
		n=2000	n=4000	n=8000	n=2000	n=4000	n=8000
EM	None	0.018; 0.38	0.013; 0.48	0.012; 0.76	**0.047; 0.05**	**0.045; 0.19**	0.044; 0.52
BBSE-soft	None	0.021; 0.62	0.014; 0.52	**0.01; 0.24**	0.08; 0.95	0.064; 0.81	0.045; 0.48
EM	TS	0.02; 0.49	0.015; 0.55	0.014; 0.73	**0.036; 0.02**	**0.034; 0.07**	**0.032; 0.24**
BBSE-soft	TS	0.021; 0.51	0.014; 0.45	**0.01; 0.27**	0.078; 0.98	0.062; 0.93	0.043; 0.76
EM	NBVS	**0.016; 0.28**	**0.01; 0.3**	**0.008; 0.38**	**0.033; 0.02**	**0.028; 0.09**	**0.026; 0.21**
BBSE-soft	NBVS	0.02; 0.72	0.014; 0.7	0.01; 0.62	0.075; 0.98	0.058; 0.91	0.04; 0.79
EM	BCTS	**0.016; 0.34**	**0.01; 0.27**	**0.009; 0.39**	**0.028; 0.0**	**0.023; 0.05**	**0.021; 0.12**
BBSE-soft	BCTS	0.02; 0.66	0.014; 0.73	0.01; 0.61	0.074; 1.0	0.057; 0.95	0.038; 0.88
EM	VS	**0.016; 0.29**	**0.01; 0.21**	**0.008; 0.27**	**0.029; 0.01**	**0.023; 0.04**	**0.02; 0.05**
BBSE-soft	VS	0.02; 0.71	0.014; 0.79	0.01; 0.73	0.074; 0.99	0.057; 0.96	0.038; 0.95

Table C.2: **CIFAR10: Comparison of EM and BBSE at correcting for "tweak-one" shift**. Metric is JS Divergence. Analogous to **Table 8**.

D CIFAR100 SUPPLEMENTARY TABLES

Shift Estimator	Calibration Method	$\alpha = 0.1$			$\alpha = 1.0$		
		n=7000	n=8500	n=10000	n=7000	n=8500	n=10000
BBSE-soft	None	0.248; 3.85	0.241; 3.84	0.237; 3.94	0.21; 4.0	0.204; 4.0	0.198; 4.0
BBSE-soft	TS	**0.224; 0.9**	**0.218; 1.08**	**0.214; 1.21**	**0.187; 1.26**	**0.181; 1.46**	**0.176; 1.52**
BBSE-soft	NBVS	0.226; 2.12	0.22; 2.12	0.216; 2.04	0.189; 2.15	0.183; 2.15	0.178; 2.23
BBSE-soft	BCTS	**0.224; 1.22**	**0.218; 1.2**	**0.214; 1.11**	**0.187; 1.06**	**0.181; 0.83**	**0.176; 0.82**
BBSE-soft	VS	0.226; 1.91	0.22; 1.76	0.215; 1.7	0.188; 1.53	0.182; 1.56	0.177; 1.43

Table D.1: **CIFAR100: Comparison of calibration methods when using BBSE adaptation to dirichlet shift, with JS Divergence (Sec. 3.3) as the metric.** Analogous to **Table 1**

E DIABETIC RETINOPATHY SUPPLEMENTARY TABLES

Shift Estimator	Calibration Method	$\rho = 0.5$			$\rho = 0.9$		
		n=500	n=1000	n=1500	n=500	n=1000	n=1500
BBSE-soft	None	0.17; 2.08	0.127; 2.51	0.105; 2.21	0.131; 2.89	0.096; 2.93	0.091; 2.82
BBSE-soft	TS	**0.16; 1.86**	**0.119; 1.86**	**0.1; 1.9**	0.122; 2.52	0.092; 2.53	0.089; 2.57
BBSE-soft	NBVS	0.176; 2.32	**0.121; 2.05**	0.106; 2.32	0.125; 2.26	0.093; 2.18	0.086; 2.13
BBSE-soft	BCTS	**0.167; 1.9**	**0.116; 1.63**	**0.096; 1.85**	0.107; 1.32	**0.079; 1.13**	**0.077; 1.17**
BBSE-soft	VS	**0.159; 1.84**	**0.118; 1.95**	**0.093; 1.72**	**0.101; 1.01**	**0.079; 1.23**	**0.077; 1.31**

Table E.1: **Kaggle Diabetic Retinopathy: Comparison of calibration methods when using BBSE adaptation to domain shift, with JS Divergence (Sec. 3.3) as the metric.** Analogous to **Table 1**. See **Sec. 4.1** for details.

Calibration Method	NLL			ECE		
	n=500	n=1000	n=1500	n=500	n=1000	n=1500
None	0.64; 4.0	0.639; 4.0	0.639; 4.0	8.734; 4.0	8.737; 4.0	8.767; 4.0
TS	0.571; 3.0	0.57; 3.0	0.569; 3.0	3.65; 2.77	3.729; 2.92	3.853; 2.76
NBVS	0.543; 2.0	0.54; 2.0	0.539; 2.0	**2.13; 0.67**	**2.028; 0.97**	**2.129; 1.01**
BCTS	**0.514; 0.21**	**0.511; 0.57**	0.511; 0.63	2.255; 1.21	**2.097; 1.17**	**2.171; 1.14**
VS	0.518; 0.79	**0.512; 0.43**	**0.51; 0.37**	2.323; 1.35	**2.065; 0.94**	**2.153; 1.09**

Table E.2: **Kaggle Diabetic Retinopathy: NLL and ECE for different calibration methods**. Analogous to **Table 6**.

Shift Estimator	Calibration Method	$\rho = 0.5$			$\rho = 0.9$		
		n=500	n=1000	n=1500	n=500	n=1000	n=1500
EM	None	**0.077; 0.04**	**0.059; 0.05**	**0.054; 0.09**	**0.111; 0.4**	0.1; 0.5	0.102; 0.59
BBSE-soft	None	0.17; 0.96	0.127; 0.95	0.105; 0.91	0.131; 0.6	0.096; 0.5	**0.091; 0.41**
EM	TS	**0.09; 0.06**	**0.068; 0.07**	**0.061; 0.05**	**0.104; 0.37**	0.094; 0.43	0.094; 0.51
BBSE-soft	TS	0.16; 0.94	0.119; 0.93	0.1; 0.95	0.122; 0.63	0.092; 0.57	0.089; 0.49
EM	NBVS	**0.107; 0.12**	**0.089; 0.18**	**0.079; 0.16**	0.11; 0.47	0.1; 0.54	0.102; 0.62
BBSE-soft	NBVS	0.176; 0.88	0.121; 0.82	0.106; 0.84	0.125; 0.53	0.093; 0.46	**0.086; 0.38**
EM	BCTS	**0.111; 0.22**	**0.095; 0.24**	**0.082; 0.34**	**0.078; 0.22**	**0.065; 0.29**	**0.063; 0.26**
BBSE-soft	BCTS	0.167; 0.78	0.116; 0.76	0.096; 0.66	0.107; 0.78	0.079; 0.71	0.077; 0.74
EM	VS	**0.108; 0.17**	**0.088; 0.2**	**0.076; 0.28**	**0.077; 0.18**	**0.062; 0.22**	**0.059; 0.25**
BBSE-soft	VS	0.159; 0.83	0.118; 0.8	0.093; 0.72	0.101; 0.82	0.079; 0.78	0.077; 0.75

Table E.3: **Kaggle Diabetic Retinopathy: Comparison of EM and BBSE at correcting for domain shift**. Metric is JS Divergence. Analogous to **Table 8**. See **Sec. 4.1** for details.