Reproduction and Extension of Verified Uncertainty Calibration

He Li
New York University
hli@stern.nyu.edu

1 Introduction and Setups

The problem of predicting probability estimates representative of the true correctness likelihood is important for classification models in many applications. In this project, we reproduce and extend the experimental results in Kumar et al. (2019).

Let $\mathcal{X}$ be the input (feature) space and $\mathcal{Y}$ be the label space where $\mathcal{Y} = \{0, 1\}$ for binary classification and $\mathcal{Y} = [K] = \{1, \ldots, K\}$ for K-classification. Let $(X, Y) \in \mathcal{X} \times \mathcal{Y}$ be input-label random variable given by some distribution P. Suppose we have a model $f : \mathcal{X} \to [0, 1]^K$, for example, f can be a deep neural network which outputs a confidence measure for each class in $[K]$.

When $K = 2$, we have a binary classification problem and the calibration error of $f : \mathcal{X} \to [0, 1]$ is given by

$$CE_2(f) = \left(\mathbb{E}\left[\left|f(X) - \mathbb{E}[Y|f(X)]\right|^2\right]\right)^{1/2}. \tag{1}$$

We can also define ℓ_p calibration error accordingly.

Similarly, under the multi-class setting, we have several different measurement for calibration error. The top-label ℓ_2 calibration error is defined as

$$\mathrm{TCE}_2(f) = \left(\mathbb{E}\left[\left(\mathbb{P}\left(Y = \arg\max_{j\in[K]} f(X)_j \middle| \max_{j\in[K]} f(X)_j\right) - \max_{j\in[K]} f(X)_j\right)^2\right]\right)^{1/2}. \tag{2}$$

If we would like the model to be calibrated on all of these predictions, we can define the marginal ℓ_2 calibration error as follows,

$$\mathrm{MCE}_2(f) = \left(\sum_{j=1}^{K} \omega_k \mathbb{E}\left[\left(\mathbb{P}\left(Y = j \middle| f(X)_j\right) - f(X)_j\right)^2\right]\right)^{1/2}. \tag{3}$$

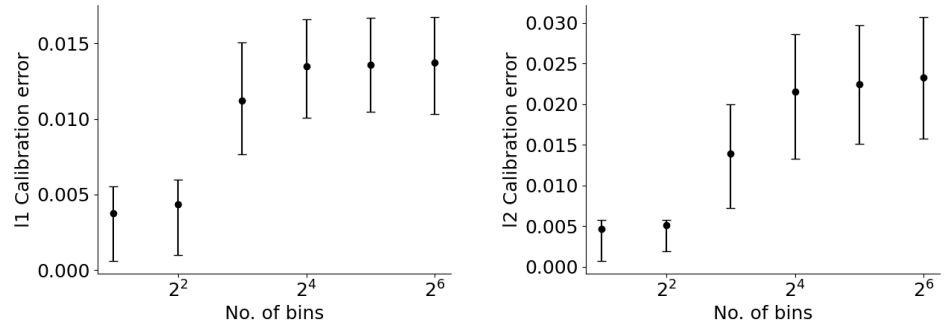
Define $\Delta_K = \{p : p \in [0, 1]^d, \|p\|_1 = 1\}$ as the simplex in $\mathbb{R}^K$, the calibration aims at finding some function $g : \Delta_k \to \Delta_k$ such that $g \circ f$ has a small calibration error.

Given a data point (X_i, y_i) with uncalibrated probability p_i and logits vector z_i from the last layer of neural network, there are some classical methods in literature to produce a calibrated probability q_i. For binary classification, **Histogram binning** (Zadrozny and Elkan, 2001) first defines bin boundaries $0 = a_1 \le a_2 \le \cdots \le a_{M+1} = 1$, it considers the following optimization problem:

$$\min_{\theta_1,\ldots,\theta_M} \sum_{m=1}^{M} \sum_{i=1}^{n} \mathbf{1}\left(a_m \le p_i \le a_{m+1}\right)(\theta_m - y_i)^2. \tag{4}$$

Given fixed bins boundaries, the solution to (4) results in θ_m that correspond to the average number of positive-class samples in its corresponding bin. It can be extended to multi-classification by treating the problem as K one-versus-all problems.

DS.GA-3001 Tools and Techniques for Machine Learning, Spring 2021.



(a) ℓ_1 calibration error

(b) ℓ_2 calibration error

Figure 1: Binned calibration errors of a recalibrated VGG16 model on CIFAR-10 with 90% confidence intervals.

Guo et al. (2017) propose **Temperature scaling** which can be seen as an extension of Platt scaling. Using a temperature parameter $T > 0$, given logits vector z_i, we output

$$q_i^{(j)} = \frac{\exp(z_i^{(j)}/T)}{\sum \exp(z_i^{(k)}/T)}, \tag{5}$$

for $q_i = (q_i^{(1)}, \ldots, q_i^{(K)})^\top$. The temperature parameter T can be optimized on validation set using negative log-likelihood.

Similar to the temperature scaling, Platt et al. (1999) considers the Platt scaling method using linear transformation. For simplicity, we would use f to denote the calibrated output function $g \circ f$ below.

2 Is Platt Scaling Calibrated?

Methods like temperature scaling and Platt scaling produce a continuous function $f(\cdot)$ and prior work bins the output of f into B intervals to approximate the calibration error, where we use the average function value within each bin to evaluate the calibration error. Denote the binned version of f as $f_{\mathcal{B}}$,

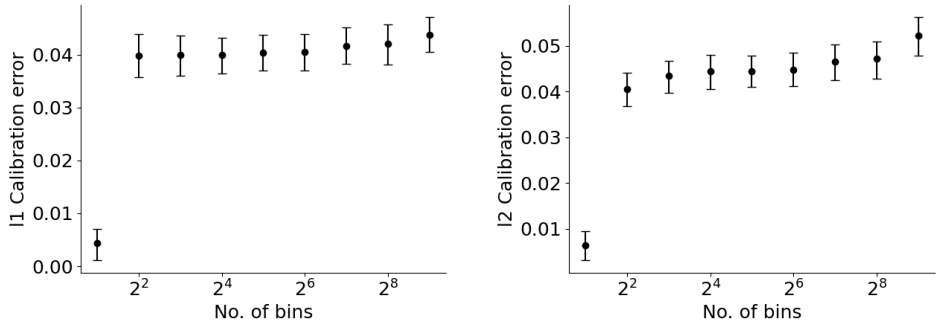
$$f_{\mathcal{B}}(x) = \mathbb{E}\left[f(X) | f(X) \in I_j\right], \;\; x \in I_j, \tag{6}$$

where $\mathcal{B} = \bigcup_j I_j$ is the partition of $[0, 1]$. As proved in Kumar et al. (2019), the calibration error of the binned version $f_{\mathcal{B}}$ is always the lower bound of the true calibration error of f (Kumar et al., 2019, Proposition 3.3).

We reproduce the experiments in Kumar et al. (2019) to verify the theoretical findings above. Our experiments on CIFAR-10 dataset has a validation set of size 10000. We split it into three sets with sizes $(1000, 1000, 8000)$ where the first set of 1000 data is used to calibrate the neural network's output using Platt scaling method, the second set to select the binning partition scheme $\mathcal{B}$ and each bin contains approximately equal number of data. Finally, we use the last 8000 data to evaluate the calibration error for the binned function $f_{\mathcal{B}}$.

For the ImageNet dataset with a validation set of size 50000, our split plan is $(20000, 5000, 25000)$. We consider a trained VGG16 model for both datasets as the uncalibrated model where the test accuracy on CIFAR-10 is 93.1% and 64.3%, respectively. In addition to the ℓ_2 top calibration error reported in the original paper, we also consider the ℓ_1 top calibration error as well. We run 1000 bootstrap resamples and report the 90% confidence interval for the calibration error. As indicated in Figure 1 and 2 below, as we increase the number of bins on both datasets, the approximated calibration errors get larger.

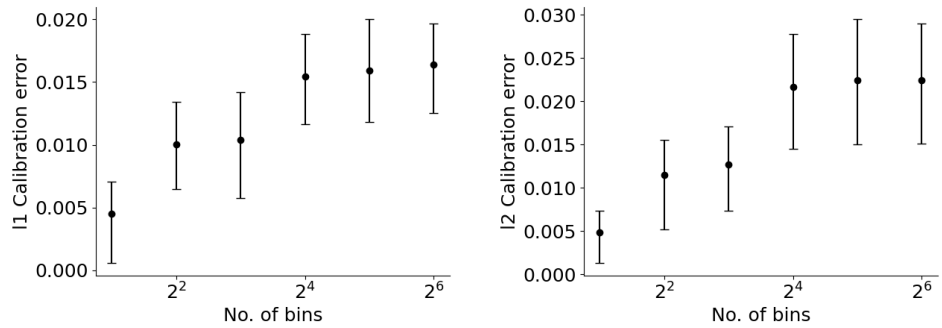
To further check our conclusion, we rerun the test on CIFAR-10 dataset with the same setting, under two different uncalibrated model, a ResNet18 model with 93.07% test accuracy and a DenseNet121 with 94.06% test accuracy. As shown in Figure 3 below, we obtain the same conclusion.



(a) ℓ_1 calibration error

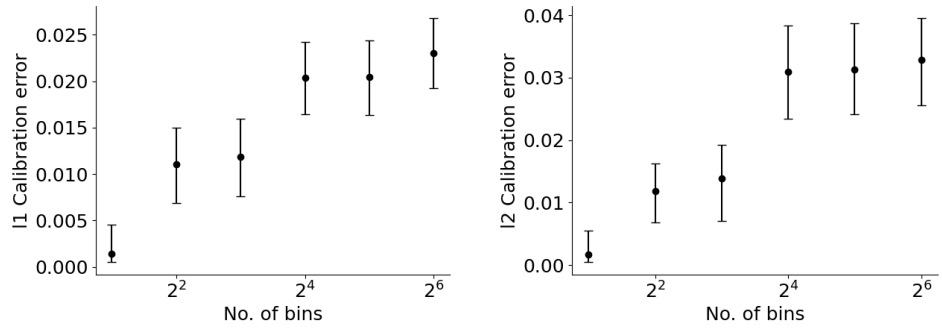
(b) ℓ_2 calibration error

Figure 2: Binned calibration errors of a recalibrated VGG16 model on ImageNet with 90% confidence intervals.



(a) ResNet18 with ℓ_1 calibration error

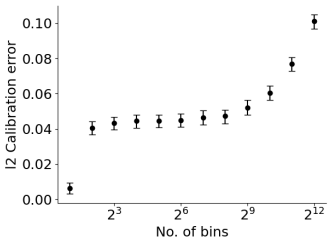
(b) ResNet18 with ℓ_2 calibration error



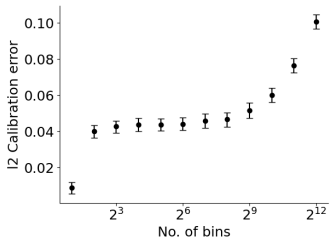
(c) DenseNet121 with ℓ_1 calibration error

(d) DenseNet121 with ℓ_2 calibration error

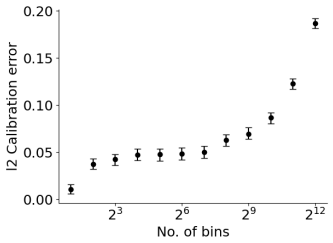
Figure 3: Binned calibration errors on CIFAR-10 with 90% confidence intervals.



(a) Split $(20000, 5000, 25000)$



(b) Split $(15000, 10000, 25000)$



(c) Split $(20000, 20000, 10000)$

Figure 4: The ℓ_2 binned calibration errors on ImageNet with different dataset partition. We use tuple (T_1, T_2, T_3) to describe the partition using T_1 data as calibration set for the Platt scaling calibrator, T_2 data for the binning scheme and T_3 data for calibration error calculation.

In their original experiments, the calibration error on ImageNet seems to be significantly increasing at 2^9 bins. We further check this fact using more bins. We consider the maximum number of bins to be 2^{12} and as can be inferred from Figure 4, the increase in the empirical calibration error is most likely caused by the lack of samples within each bin.

3 The Scaling-binning Calibrator

Kumar et al. (2019) propose a new method, which combines the idea of Platt scaling and histogram binning. Specifically, **the scaling-binning calibrator** split the recalibration set into T_1, T_2, T_3 subsets. It first fits a function g such that

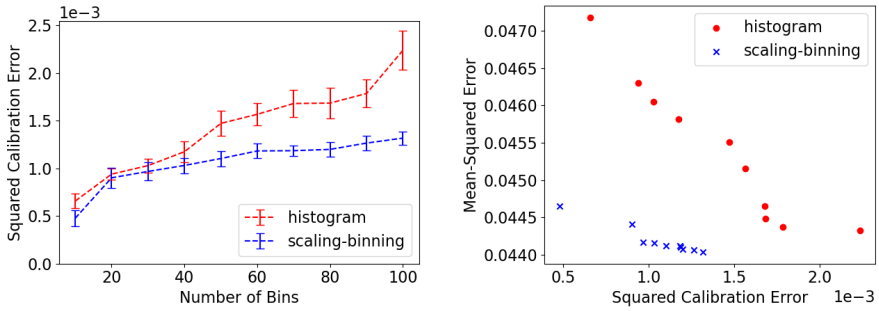
$$g = \arg\min_{g \in \mathcal{G}} \sum_{(z,y) \in T_1} (y - g(z))^2,$$

for some functional space $\mathcal{G}$. In the original paper, the authors consider the Platt scaling method for $\mathcal{G}$. Then it choose the bins so that an equal number of $g(z_i)$ in T_2 land in each bin. Finally, the function g is discretize by outputting the average g value in each bin.

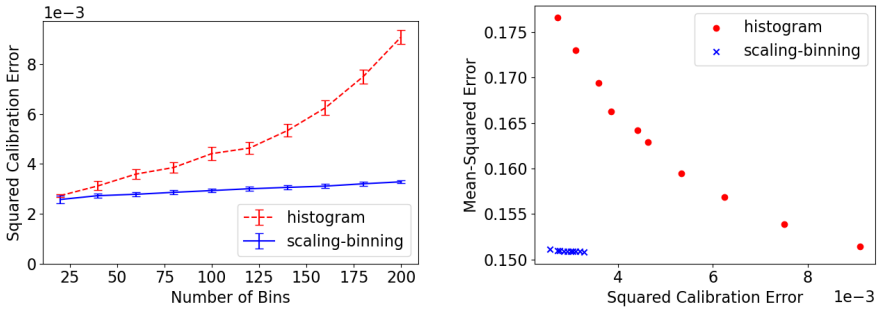
In their Theorem 4.1, Kumar et al. (2019) show that the scaling-binning calibrator is sample-efficient theoretically, better than histogram binning methods. Specifically, given some calibration error threshold $\varepsilon > 0$, the scaling-binning calibrator requires $\mathcal{O}(B + 1/\varepsilon^2)$ samples to calibrate where B is the number of bins we use for the binning partition scheme on T_2 above. Meanwhile, the histogram binning method requires a sample complexity of $\mathcal{O}(B + B/\varepsilon^2)$ to achieve the same error threshold.

We now deliver the experiments to verify this result. On both ImageNet and CIFAR-10 datasets, we sample with replacement, a recalibration set of 1000 points. This recalibration set is used to fit the scaling-binning calibrator or the histogram binning calibrator. The calibration error for both methods are evaluated on the entire validation set.

In Figure 5, plot (a) and (c) shows that when we fix the sample size for the calibration stage, the scaling-binning calibrator produces models with lower calibration error than histogram binning calibrator, on both CIFAR-10 and ImageNet datasets. It indicates that given some calibration error

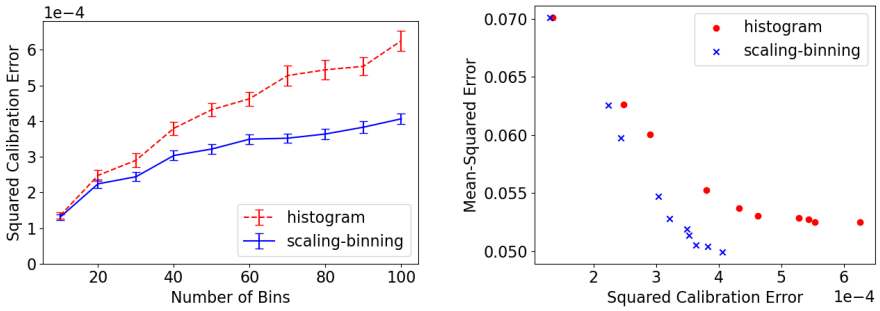


(a) TCE on CIFAR-10 (b) TCE versus MSE on CIFAR-10



(c) TCE on ImageNet (d) TCE versus MSE on ImageNet

Figure 5: Top calibration error (TCE) with 90% confidence intervals using VGG16 model.



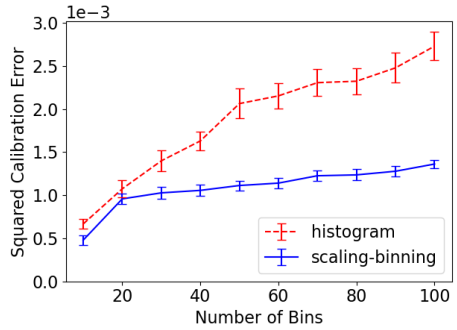
(a) MCE on CIFAR-10 (b) MCE versus MSE on CIFAR-10

Figure 6: Marginal calibration error (MCE) with 90% confidence intervals on CIFAR-10 using VGG16 model.

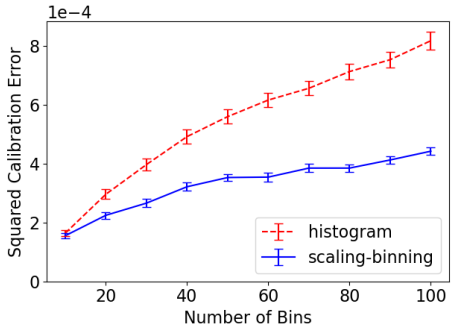
threshold, the scaling-binning calibrator can use more bins in the calibration stage, which results in a lower mean-squared error and stronger predictive power (shown in plot (b) and (d)).

We conduct experiments on the CIFAR-10 dataset using the ℓ_2 marginal calibration error as measurement. The result is presented in Figure 6 below. Similar results also apply to the ResNet18 and Densenet121 models in Figure 7.

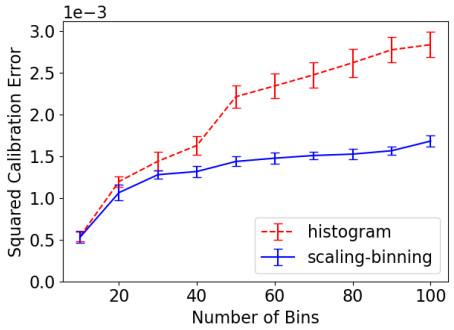
We extend the experiment by comparing different functional space $\mathcal{G}$ in the first stage of the scaling-binning calibrator. Specifically, we further consider the temperature-scaling method (Guo et al., 2017) and Dirichlet calibrator proposed in Kull et al. (2019). The result is given in Figure 8. The Dirichlet calibrator does not perform well for the scaling-binning calibrator and the Platt scaling is slightly better than the temperature scaling method given it introduces an additional bias parameter.



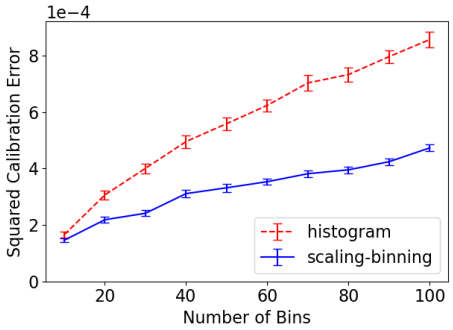
(a) TCE on CIFAR-10 with ResNet18



(b) MCE on CIFAR-10 with ResNet18



(c) TCE on CIFAR-10 with Densenet121



(d) MCE on CIFAR-10 with Densenet121

Figure 7: Calibration error with 90% confidence intervals on CIFAR-10 using different models.

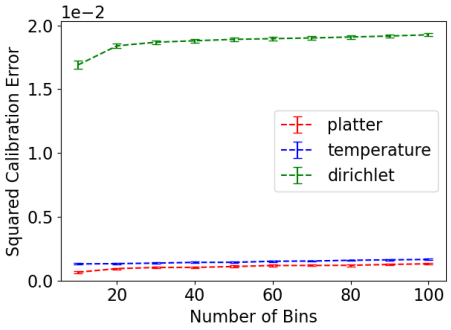
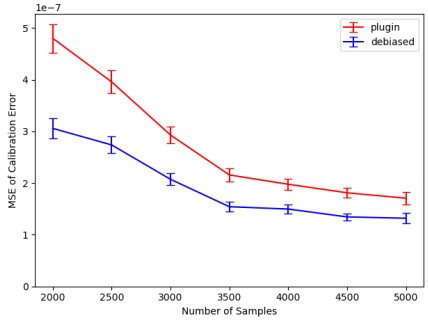


Figure 8: Top calibration error with 90% confidence intervals by VGG16-scaling-binning based model on CIFAR-10.

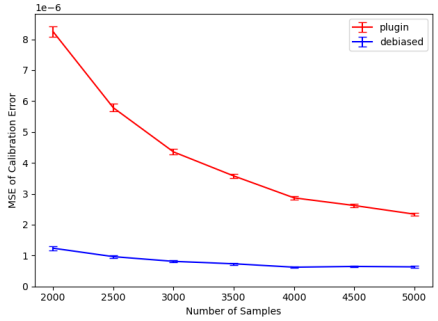
4 Debiased Calibration Error Estimator

For the binary classification case, the calibration error is defined in Equation (1). For a binned model like the scaling-binning calibrator and the histogram binning method, the natural way of estimating the calibration error empirically is by plugin samples directly. Denote $T_n = \{(x_1, y_1), \ldots, (x_n, y_n\}$ as the evaluation set for model f. Denote $L_s = \{y_j | (x_j, y_j) \in T_n,\ f(x_j) = s\}$ as the set of labels y_j where the model output value is s. The plugin estimates for the ℓ_2 calibration error is given by

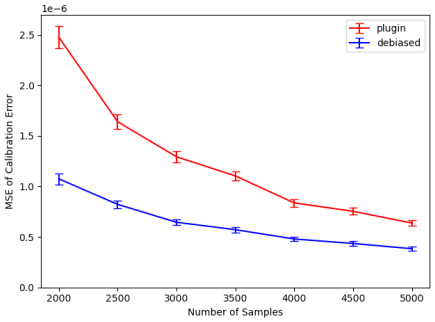
$$\widehat{CE}(f) = \sqrt{\sum_s \hat{p}_s (s - \hat{y}_s)^2}. \tag{7}$$



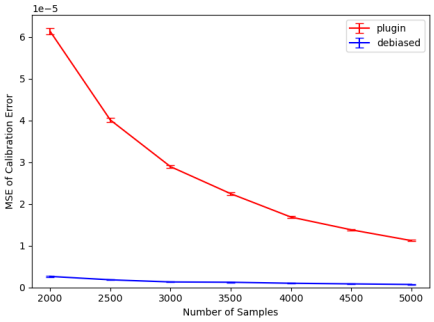
(a) CIFAR-10 with 15 bins



(b) CIFAR-10 with 100 bins

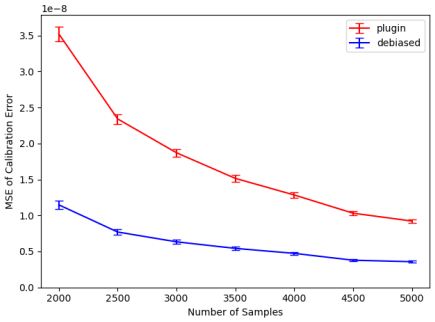


(c) ImageNet with 15 bins

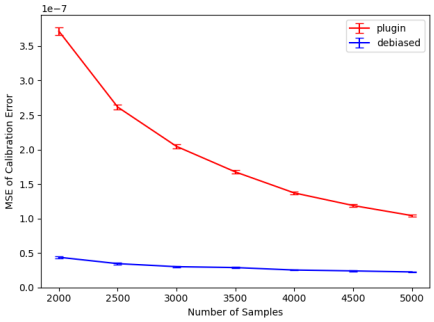


(d) ImageNet with 100 bins

Figure 9: Comparison between the debiased estimator and the plugin estimator. We estimate the TCE here and vary the number of bins for the scaling-binning calibrator.

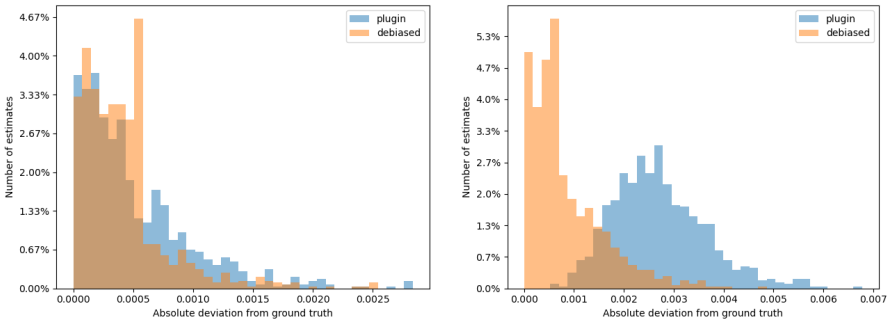


(a) CIFAR-10 with 15 bins



(b) CIFAR-10 with 100 bins

Figure 10: Comparison between the debiased estimator and the plugin estimator on the CIFAR-10 dataset. We estimate the MCE here and vary the number of bins for the scaling-binning calibrator.



(a) CIFAR-10 with 15 bins (b) CIFAR-10 with 100 bins

Figure 11: Comparison between the debiased estimator and the plugin estimator on the CIFAR-10 dataset. We estimate the MCE here and vary the number of bins for the scaling-binning calibrator.

where $\hat{p}_s$ is the estimated probability for L_s: $\hat{p}_s = |L_s|/n$ and $\hat{y}_s$ is the empirical average of labels within the set L_s: $\hat{y}_s = \frac{1}{|L_s|}\sum_{y_j \in L_s} y_j$.

In their paper, Kumar et al. (2019) propose a debiased estimator which has more estimation accuracy than the plugin estimator given the same number of evaluation samples (Theorem 5.3 and Theorem 5.4). The debiased estimator has the following form,

$$\widetilde{CE}(f) = \sqrt{\sum_s \hat{p}_s \left[(s - \hat{y}_s)^2 - \frac{\hat{y}_s(1-\hat{y}_s)}{\hat{p}_s n - 1}\right]}. \tag{8}$$

We now describe the details of our experiments for this debiased estimator. On both the CIFAR-10 and ImageNet datasets, we use 3000 samples as the calibration set for our scaling-binning estimator and the rest as the evaluation set. We sample a proportion of the evaluation set to compare the debiased estimator and the plugin estimator and the performance for the two estimators are measured by the mean-squared error. Our experiment is based on 1000 bootstrap resamples and 90% confidence intervals are constructed. In Figure 9, the debiased estimator of the top calibration error is much closer to the ground truth than the plugin estimator on both datasets. We extend the result to the marginal calibration error in Figure 10 on CIFAR-10. In Figure 11, we provide the distribution of the 1000 bootstrap resamples for the TCE case on CIFAR-10, which also shows the debiased estimator is superior than the plugin estimator. Similar results can be found on the ImageNet as well. We also conduct experiments on the ResNet18 and DenseNet121 structures. We omit those results as they all suggest the same conclusion.

5 Conclusion

In this project, we reproduce and extend the results in Kumar et al. (2019). Mainly, we show the following three conclusions with our experiments on the real world datasets.

1. The calibration errors of the continuous methods are underestimated using binning partition.
2. The scaling-binning calibrator is proposed which has a lower sample complexity than the histogram binning calibrator and also has a measurable calibration error empirically.
3. The debiased estimator for calibration error is presented. It has a lower sample complexity than the classical plugin estimator.

References

Guo, C., G. Pleiss, Y. Sun, and K. Q. Weinberger (2017). On calibration of modern neural networks. In *International Conference on Machine Learning*, pp. 1321–1330. PMLR.

Kull, M., M. Perello Nieto, M. Kängsepp, T. Silva Filho, H. Song, and P. Flach (2019). Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration. In *Advances in Neural Information Processing Systems*.

Kumar, A., P. S. Liang, and T. Ma (2019). Verified uncertainty calibration. In *Advances in Neural Information Processing Systems*.

Mukhoti, J., V. Kulharia, A. Sanyal, S. Golodetz, P. Torr, and P. Dokania (2020). Calibrating deep neural networks using focal loss. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc.

Platt, J. et al. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers 10*(3), 61–74.

Zadrozny, B. and C. Elkan (2001). Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *Icml*, Volume 1, pp. 609–616. Citeseer.

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