Mitigating Bias in Calibration Error Estimation

Rebecca Roelofs  
Google Research

Nicholas Cain  
Google Research

Jonathon Shlens  
Google Research

Michael C. Mozer  
Google Research

Abstract

For an AI system to be reliable, the confidence it expresses in its decisions must match its accuracy. To assess the degree of match, examples are typically binned by confidence and the per-bin mean confidence and accuracy are compared. Most research in calibration focuses on techniques to reduce this empirical measure of calibration error, $ECE_{\text{bin}}$. We instead focus on assessing statistical bias in this empirical measure, and we identify better estimators. We propose a framework through which we can compute the bias of a particular estimator for an evaluation data set of a given size. The framework involves synthesizing model outputs that have the same statistics as common neural architectures on popular data sets. We find that binning-based estimators with bins of equal mass (number of instances) have lower bias than estimators with bins of equal width. Our results indicate two reliable calibration-error estimators: the debiased estimator (Brocker, 2012; Ferro and Fricker, 2012) and a method we propose, $ECE_{\text{sweep}}$, which uses equal-mass bins and chooses the number of bins to be as large as possible while preserving monotonicity in the calibration function. With these estimators, we observe improvements in the effectiveness of recalibration methods and in the detection of model miscalibration.

1 INTRODUCTION

Machine learning models are increasingly deployed in high-stakes settings like self-driving cars (Caesar et al., 2020; Geiger et al., 2013; Sun et al., 2020) and medical diagnosis (Esteva et al., 2017, 2019; Gulshan et al., 2016), where it is critical to recognize when a model is likely to be incorrect. Unfortunately, models often fail in unexpected and poorly understood ways, hindering our ability to interpret and trust such systems (Azulay and Weiss, 2018; Biggio and Roli, 2018; Hendrycks and Dietterich, 2019; Recht et al., 2019; Szegedy et al., 2013). To address these issues, calibration is used to ensure that a model produces confidence scores that reflect its ground truth likelihood of being correct (Platt et al., 1999; Zadrozny and Elkan, 2001, 2002).

To obtain an estimate of the calibration error, or $ECE_{\text{bin}}$, the standard procedure partitions the model confidence scores into bins and compares the model’s predicted accuracy to its empirical accuracy within each bin (Guo et al., 2017; Naeini et al., 2015). We refer to this specific metric as $ECE_{\text{bin}}$. Recent work observed that the calculation of $ECE_{\text{bin}}$ is sensitive to implementation (Kumar et al., 2019; Nixon et al., 2019). Fundamentally, a key confounding factor is statistical bias, the difference between the expected $ECE_{\text{bin}}$ and the true calibration error (TCE). Because bias is largely unexplored in the literature, its magnitude and sign is unknown, as is its dependence on hyperparameters of the $ECE_{\text{bin}}$ estimator (e.g., number of bins, how bins are formed). We explain our reasons for focusing on estimator bias and not variance in Section 4.

Bias in $ECE_{\text{bin}}$ measurement has two real world consequences. First, the measurement of calibration error on a given model may be systematically incorrect. Thus, our understanding of how well a model knows whether it is correct may be poor, and may not be accurately captured by naively reporting $ECE_{\text{bin}}$. Second, many techniques have been developed to minimize the calibration error, such as post-hoc recalibration techniques (Guo et al., 2017; Zadrozny and Elkan, 2001, 2002) and, more recently, calibration-sensitive training objectives (Karandikar et al., 2021; Krishnan and Tickoo, 2020; Kumar et al., 2018; Lin et al., 2018; Mukhoti et al., 2020). Given that the selection of the training objectives and the justification of a recalibration technique is predicated on the measurement of the calibration

---

\[\text{Naeini et al. (2015) introduce ECE as an acronym for Expected Calibration Error. However, ECE is not a proper expectation whereas the true calibration error is computed under an expectation. To resolve this confusion, we prefer to read ECE as Estimated Calibration Error.}\]
Mitigating Bias in Calibration Error Estimation

(a) Figure 1: ECEBIN exhibits large bias for perfectly calibrated models. We simulate data from a perfectly calibrated model with confidence scores fit to ResNet-110 CIFAR-10 output (He et al., 2016; Kangsepp, 2019) and measure ECEBIN using 15 equal-width spaced bins. The left panel shows a reliability diagram for a sample of size n = 200 (named Sample B); the right panel has a distribution of ECEBIN scores computed across 10^5 independent simulations. Even though the model is perfectly calibrated, ECEBIN systematically predicts large calibration errors.

Error, reliance on an inaccurate estimator may lead to a suboptimal choice.

We address this problem by developing a technique to measure bias in calibration metrics, which we call the bias-by-construction (BBC) framework. The BBC framework uses simulation to create a setting where the TCE can be computed analytically and thus the bias can be estimated directly. BBC reveals that ECEBIN has systematic non-negligible statistical bias, particularly for perfectly calibrated models (Figure 1).

Our goal is to identify the least biased estimator of calibration error using BBC. We consider two estimators previously proposed in the literature: the debiased estimator (Brocker, 2012; Ferro and Fricker, 2012), which we refer to as ECEDEBIAS, and the smoothed kernel density estimator of Zhang et al. (2020), which we refer to as KDE. Additionally, we propose an extension of ECEBIN where the number of bins is chosen to ensure monotonicity of the calibration histogram, which we refer to as ECESWEEP. ECEBIN, ECEDEBIAS, and ECESWEEP all require the binning of model confidence scores, and under the lens of bias, we examine two common methods for specifying bins: partitioning the confidence-score continuum either into equal width bins or bins of equal mass—equal numbers of data instances.

Furthermore, BBC allows us to examine the impact of biased estimators in downstream decision making, such as the selection of a post-hoc recalibration method. For example, when the choices for recalibration include histogram binning (Zadrozny and Elkan, 2001), temperature scaling (Guo et al., 2017), and isotonic regression (Zadrozny and Elkan, 2002), Table 1 illustrates that our bias-reduced measure, ECESWEEP, more frequently selects the ‘optimal’ recalibration method when compared to the standard measure, ECEBIN (70% versus 30% correctness, respectively). Optimality is determined by estimating TCE using numerical integration on curves arising from maximum likelihood fits across multiple model families, where we select the best model via the Akaike information criterion (see Section 6).

To summarize the contributions of this work, the core contribution is a simulation framework, bias by construction or BBC, that allows us to identify and characterize systematic bias in calibration error metrics for realistic models and data sets. We show that estimation of calibration error by the predominant method, ECEBIN, is biased, and paradoxically the bias is most severe for perfectly calibrated models. Bias can lead not only to the mis-estimation of calibration error but also to the wrong choice of recalibration method, yielding a poorly calibrated model. Moreover, we find that the selection of hyperparameters for measuring calibration (e.g., number of bins) is under-appreciated and is absolutely critical. To address these issues, we propose ECESWEEP, a simple algorithm based on the monotonicity principle of calibration curves. We compare the bias of various estimators using predictions from four popular neural architectures and three data sets. We find that ECEBIN is more biased than either ECEDEBIAS or ECESWEEP, and of these two improved measures, ECEDEBIAS performs better for perfectly calibrated models and ECESWEEP for miscalibrated models. Finally, our analyses provide rigorous empirical evidence that for all binning-based estimators, equal-mass binning obtains a more accurate estimate of true calibration error. This finding gives strong guidance to revise the current practice of equal-width binning.

2 RELATED WORK

ECEBIN. ECEBIN with 15 bins of equal width is currently the most popular way to measure calibration error in the literature (Guo et al., 2017; Naeini et al., 2015). An alternative but less popular implementation evaluates ECEBIN using bins of equal mass, which partitions examples into bins that have an equal number of examples (Kumar et al., 2019; Zadrozny and Elkan, 2001). Recently, Nixon et al. (2019) observed that ECEBIN with equal-mass binning produces more stable rankings of recalibration algorithms, which is consistent with our conclusion that equal mass ECEBIN is a less biased estimator of TCE.

Sensitivity of ECEBIN to implementation hyperparameters. Several works have pointed out that ECEBIN is sensitive to implementation details. Kumar et al. (2019) show that ECEBIN increases with number of bins. Nixon et al. (2019) find that ECEBIN is sen-
Alternative definitions of calibration error. Researchers have studied alternatives notions of calibration error that are distinct from TCE (see Section 3 for a formal definition of TCE). For example, Widmann et al. (2019) proposed a kernel-based calibration error, KCE, which has no explicit dependence on the model’s calibration function. Gupta et al. (2020) propose a calibration error metric inspired by the Kolmogorov-Smirnov (KS) statistical test that estimates the maximum difference between cumulative probability distributions describing the model’s confidence and accuracy. The KS is similar to the maximum calibration error (MCE) (Naeini et al., 2015) in that it computes a worst-case deviation between confidence and accuracy, but the KS is computed on the CDF, while the MCE uses binning and is computed on the PDF. In contrast, TCE measures the average difference between confidence and accuracy. Both the worst case and average difference are useful measures but may be applicable under different circumstances (Guo et al. 2017).

Monotonicity in calibration curves. While Zadrozny and Elkan (2002) used calibration curve monotonicity to motivate isotonic regression for recalibration, they observed monotonic calibration curves empirically on only a handful of pre-deep learning models, and without theoretical justification. In contrast, our work is the first to suggest using monotonicity to improve calibration metrics. We provide both theoretical and extensive empirical evidence that monotonic calibration curves arise in modern deep networks.

3 BACKGROUND

Consider a binary classification setup with input $X \in \mathcal{X}$, target output $Y = \{0, 1\}$, and suppose we have a model $f : X \rightarrow [0, 1]$ whose output represents a confidence score that the true label $Y$ is 1.

True calibration error (TCE). We define true calibration error as the $\ell_p$ norm difference between a model’s predicted confidence and the true likelihood of being correct:\footnote{In our experiments, we measure calibration error using the $\ell_2$ norm because it increases the sensitivity of the error metric to extremely poorly calibrated predictions, which tend to be more harmful in applications.}

$$\text{TCE}(f) = \left( \mathbb{E}_X \left[ ||f(X) - \mathbb{E}_Y[Y|f(X)]||_p \right] \right)^{1/p}. \quad (1)$$

Two independent features of a model determine TCE: (1) the distribution of confidence scores $f(x) \sim \mathcal{F}$ over which the outer expectation is computed, and (2) the true calibration curve $\mathbb{E}_Y[Y|f(X)]$, which governs the relationship between the confidence score $f(x)$ and the empirical accuracy (see Figure 2 for illustration).

3.1 Estimates of calibration error

To estimate the TCE of a model $f$, assume we are given a dataset containing $n$ samples, $\{x_i, y_i\}_{i=1}^n$. We can approximate TCE by replacing the outer expectation in Equation 1 by the sample average and replacing the

| CIFAR-10 | CIFAR-100 | ImageNet |
|----------|-----------|-----------|
| ResNet 110 | ResNet 110 SD | WideResNet 32 | DenseNet 40 | WideResNet 32 | DenseNet 152 | DenseNet 161 |
| ECE$_{bin}$ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| ECE$_{sweep}$ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |

Table 1: The selection of a recalibration method is severely affected by biased computational methods. For ten models, we report whether either ECE$_{bin}$ or ECE$_{sweep}$ select the same (✓) or different (✗) recalibration algorithm (either histogram binning, isotonic regression or temperature scaling) as would an estimate of TCE obtained from maximum likelihood fits to empirical data (see Section 6). ECE$_{bin}$ selects the same algorithm only 3/10 times, versus 7/10 for ECE$_{sweep}$, illustrating how computational bias can negatively impact recalibration.
We focus on bias rather than variance because the variance can be estimated from a finite set of samples using resampling techniques whereas the bias is an unknown quantity that reflects systematic error. For completeness, we report variance for various calibration metrics as we vary the sample size, number of bins, and binning technique in Appendix B. We find empirically that the variance is relatively insensitive to the estimation technique and number of bins.

The bias of a calibration error estimator, \( \text{ECE}_A \) for some estimation algorithm \( A \), is the difference between the estimator’s expected value with respect to the data distribution and the TCE:

\[
\text{Bias}_A = \mathbb{E}[\text{ECE}_A] - \text{TCE}.
\]

If we assume a specific confidence score distribution \( F \) and true calibration curve \( T(X) = \mathbb{E}_Y[Y \mid f(X) = c] \) (see Figure 2a for examples), we can compute the TCE by analytically or numerically evaluating the integral implicit in the outer expected value of Equation 1. We then compute a sample estimate of the bias by generating \( n \) samples \( \{f(x_1), y_1\}_{i=1}^n \) such that \( f(x_1) \sim F \) and \( \mathbb{E}_Y[Y \mid f(X) = c] := T(c) \), and computing the ECE on the sample. We repeat this process for \( m \) simulated datasets and compute the sample estimate of bias (hereafter, simply the “bias”) as the difference between the average ECE and the TCE:

\[
\text{Bias}_A(n) = \frac{1}{m} \sum_{i=1}^m \text{ECE}_A - \text{TCE}.
\]

Using this bias-by-construction (BBC) framework, we next investigate the bias in \( \text{ECE}_\text{bin} \) as a function of the number of samples \( n \) and the number of bins. We compute \( \text{ECE}_\text{bin} \) with equal width binning and we assume parametric curves for \( f(x) \) and \( \mathbb{E}_Y[Y \mid f(X)] \) that are fit to the ResNet-110 CIFAR-10 model output. (Section 6 has details on how we compute fits.)

Proposition 3.3 of Kumar et al. (2019) asserts that any binned version of calibration error systematically underestimates TCE in the limit of infinite data. However, for a finite number of samples \( n \), Figure 2b shows that \( \text{ECE}_\text{bin} \) can either overestimate or underestimate TCE and that increasing the number of bins does not always lead to better estimates of TCE. In Appendix B we show how bias and variance vary for several calibration metrics as we change the binning scheme, sample size, and number of bins. Regardless of binning scheme, for \( \text{ECE}_\text{bin} \) we find empirically that there exists a bin number for each sample size that results in the lowest estimation bias and this optimal bin count grows with the sample size. Intuitively, having a large number of bins is generally preferred because we can obtain a finer-resolution estimate of the calibration curve. However, if we have a small number of samples, setting the number of bins too high may result in a poor estimate of the calibration curve due to the low number of samples in each bin.

We focus on bias rather than variance because the variance can be estimated from a finite set of samples through resampling techniques whereas the bias is an unknown quantity that reflects systematic error. For completeness, we report variance for various calibration metrics as we vary the sample size, number of bins, and binning technique in Appendix B. We find empirically that the variance is relatively insensitive to the estimation technique and number of bins.

The bias of a calibration error estimator, \( \text{ECE}_A \) for some estimation algorithm \( A \), is the difference between the estimator’s expected value with respect to the data distribution and the TCE:

\[
\text{Bias}_A = \mathbb{E}[\text{ECE}_A] - \text{TCE}.
\]

If we assume a specific confidence score distribution \( F \) and true calibration curve \( T(X) = \mathbb{E}_Y[Y \mid f(X) = c] \) (see Figure 2a for examples), we can compute the TCE by analytically or numerically evaluating the integral implicit in the outer expected value of Equation 1.

We then compute a sample estimate of the bias by generating \( n \) samples \( \{f(x_1), y_1\}_{i=1}^n \) such that \( f(x_1) \sim F \) and \( \mathbb{E}_Y[Y \mid f(X) = c] := T(c) \), and computing the ECE on the sample. We repeat this process for \( m \) simulated datasets and compute the sample estimate of bias (hereafter, simply the “bias”) as the difference between the average ECE and the TCE:

\[
\text{Bias}_A(n) = \frac{1}{m} \sum_{i=1}^m \text{ECE}_A - \text{TCE}.
\]

Using this bias-by-construction (BBC) framework, we next investigate the bias in \( \text{ECE}_\text{bin} \) as a function of the number of samples \( n \) and the number of bins. We compute \( \text{ECE}_\text{bin} \) with equal width binning and we assume parametric curves for \( f(x) \) and \( \mathbb{E}_Y[Y \mid f(X)] \) that are fit to the ResNet-110 CIFAR-10 model output. (Section 6 has details on how we compute fits.)

Proposition 3.3 of Kumar et al. (2019) asserts that any binned version of calibration error systematically underestimates TCE in the limit of infinite data. However, for a finite number of samples \( n \), Figure 2b shows that \( \text{ECE}_\text{bin} \) can either overestimate or underestimate TCE and that increasing the number of bins does not always lead to better estimates of TCE. In Appendix B we show how bias and variance vary for several calibration metrics as we change the binning scheme, sample size, and number of bins. Regardless of binning scheme, for \( \text{ECE}_\text{bin} \) we find empirically that there exists a bin number for each sample size that results in the lowest estimation bias and this optimal bin count grows with the sample size. Intuitively, having a large number of bins is generally preferred because we can obtain a finer-resolution estimate of the calibration curve. However, if we have a small number of samples, setting the number of bins too high may result in a poor estimate of the calibration curve due to the low number of samples in each bin.

4 THE BBC FRAMEWORK

We focus on bias rather than variance because the variance can be estimated from a finite set of samples through resampling techniques whereas the bias is an unknown quantity that reflects systematic error. For completeness, we report variance for various calibration metrics as we vary the sample size, number of bins, and binning technique in Appendix B. We find empirically that the variance is relatively insensitive to the estimation technique and number of bins.

The bias of a calibration error estimator, \( \text{ECE}_A \) for some estimation algorithm \( A \), is the difference between the estimator’s expected value with respect to the data distribution and the TCE:

\[
\text{Bias}_A = \mathbb{E}[\text{ECE}_A] - \text{TCE}.
\]

If we assume a specific confidence score distribution \( F \) and true calibration curve \( T(X) = \mathbb{E}_Y[Y \mid f(X) = c] \) (see Figure 2a for examples), we can compute the TCE by analytically or numerically evaluating the integral implicit in the outer expected value of Equation 1.

We then compute a sample estimate of the bias by generating \( n \) samples \( \{f(x_1), y_1\}_{i=1}^n \) such that \( f(x_1) \sim F \) and \( \mathbb{E}_Y[Y \mid f(X) = c] := T(c) \), and computing the ECE on the sample. We repeat this process for \( m \) simulated datasets and compute the sample estimate of bias (hereafter, simply the “bias”) as the difference between the average ECE and the TCE:

\[
\text{Bias}_A(n) = \frac{1}{m} \sum_{i=1}^m \text{ECE}_A - \text{TCE}.
\]

Using this bias-by-construction (BBC) framework, we next investigate the bias in \( \text{ECE}_\text{bin} \) as a function of the number of samples \( n \) and the number of bins. We compute \( \text{ECE}_\text{bin} \) with equal width binning and we assume parametric curves for \( f(x) \) and \( \mathbb{E}_Y[Y \mid f(X)] \) that are fit to the ResNet-110 CIFAR-10 model output. (Section 6 has details on how we compute fits.)

Proposition 3.3 of Kumar et al. (2019) asserts that any binned version of calibration error systematically underestimates TCE in the limit of infinite data. However, for a finite number of samples \( n \), Figure 2b shows that \( \text{ECE}_\text{bin} \) can either overestimate or underestimate TCE and that increasing the number of bins does not always lead to better estimates of TCE. In Appendix B we show how bias and variance vary for several calibration metrics as we change the binning scheme, sample size, and number of bins. Regardless of binning scheme, for \( \text{ECE}_\text{bin} \) we find empirically that there exists a bin number for each sample size that results in the lowest estimation bias and this optimal bin count grows with the sample size. Intuitively, having a large number of bins is generally preferred because we can obtain a finer-resolution estimate of the calibration curve. However, if we have a small number of samples, setting the number of bins too high may result in a poor estimate of the calibration curve due to the low number of samples in each bin.
5 MONOTONIC CALIBRATION METRICS

Though Section [4] shows that there exists an optimal number of bins for which \(ECE_{\text{bin}}\) has the lowest bias, unfortunately, this number depends on the binning technique, the number of samples, the confidence score distribution, and the true calibration curve. This observation motivates us to seek a method for adaptively choosing the number of bins.

Monotonicity in the true calibration curve implies that a model’s expected accuracy should not decrease as the model’s confidence increases. Although this requirement seems reasonable for any statistical model, it is not obvious how to prove why or when a “reasonable” model would attain such a property. We offer a rationale for why it should be expected of machine learning models trained with a maximum likelihood objective, e.g., cross-entropy or logistic loss. Namely, from ROC (receiver operating characteristic) analysis of maximum likelihood models, an under-appreciated observation of ROC curves is that a model trained to maximize the likelihood ratio must have a convex ROC curve in the limit of infinite data (see [Green et al., 1966] Sec. 2.3). The slope of the ROC curve is related to the calibration curve, and a convex ROC curve implies a monotonically increasing calibration curve (the converse is also true) ([Chen et al., 2018] [Gneiting and Vogel, 2018]).

In practice, several potential confounds may lead to observing non-monotonic calibration curves. First, finite data size may lead to fluctuations in the true positive or false positive rates, but do not reflect the behavior of the underlying model. Second, deviations in domain statistics between cross-validated splits in the data may lead to unbounded calibration errors; however, we assume that such domain shifts are negligible as cross-validated splits are presumed to be selected i.i.d. [3] Given that deviations from non-monotonic calibration curves are considered artificial, we posit that any method that is trying to assess the TCE of an underlying model may freely assume monotonicity in the true calibration curve. Note that this proposition already guides the entire field of re-calibration to require that re-calibration methods only consider monotonic functions ([Platt et al., 1999] [Wu et al., 2012] [Zadrozny and Elkan, 2002]).

Accordingly, we leverage underlying monotonicity in the true calibration curve and propose the monotonic sweep calibration error, a metric that chooses the largest number of bins possible such that the chosen bin size and all smaller bin sizes preserve monotonicity in the bin heights \(\bar{y}_k\), i.e.,

\[
ECE_{\text{sweep}} = \left( \sum_{k=1}^{b^*} \frac{|B_k|}{n} \left| \bar{f}_k - \bar{y}_k \right|^p \right)^{\frac{1}{p}}
\]

where

\[
b^* = \max\{b \mid 1 \leq b \leq n; \forall b' \leq b, \ \bar{y}_1 \leq \ldots \leq \bar{y}_{b'}\}
\]

**Algorithm 1** Monotonic Sweep Calibration Error

```plaintext
for b = 2 to n do
  Compute bin heights (\(\bar{y}_k\)) for ECEbin using b bins
  if binning is not monotonic then
    b ← b - 1
    break
  end if
return ECEbin computed with b bins
```

We compute the monotonic sweep calibration error by starting with \(b = 2\) bins (\(b = 1\) is guaranteed to be a monotonic binning) and gradually increasing the number of bins until we either reach a non-monotonic binning, in which case we return the last \(b\) that corresponded to a monotonic binning, or until every sample belongs to its own bin (\(b = n\)). In Appendix D, we explore the number of bins chosen by ECE_{sweep} for varying sample sizes and model output.

6 FITTING THE CALIBRATION CURVE AND SCORE DISTRIBUTION

TCE is analytically computable when we assume parametric forms for the confidence distribution and the true calibration curve. In order to ensure that the parametric forms we use in simulation reflect the diversity and complexity of realistic model output, we develop parametric models of empirical logit datasets.

We consider 10 publicly available logit datasets ([Kängsepp, 2019]) that arise from training four different architectures (ResNet, ResNet-SD, Wide-ResNet, and DenseNet) ([He et al., 2016] [Huang et al., 2017] [2016] [LeCun et al., 1998] [Zagoruyko and Komodakis, 2016]) on three different image datasets (CIFAR-10/100 and ImageNet) ([Deng et al., 2009] [Krizhevsky and Hinton, 2009]). For each example in a given dataset, we compute top-label confidence scores by selecting the maximum softmax score across all classes and we compute whether or not the example resulted in a “hit,” i.e. whether the model’s predicted class corresponds to the true class. By using only the top-label confidence score and determining whether the top and true labels match, we can treat the calibration problem as binary.

For the parametric fits, we model confidence score distributions \(f(X)\) using a beta density fit via maximum
likelihood estimation. The beta distribution is a flexible continuous probability distributions on the interval [0,1], which makes it a natural choice for representing the probability distribution defined by the model output. For calibration curves, we fit multiple (binary) generalized linear models (GLM) to the top-label output and then select the best model using the Akaike Information Criteria (AIC). The AIC is a standard procedure for model selection in the literature, for selecting the model that most adequately describes data arising from a mechanism included in the model family. The GLM models considered include logit, log, and "logflip" (log(1 - x)) link and transformation functions, up to first order in the transformed domain, which all result in monotonic calibration functions. See Appendix A for additional details.

We find that the parametric forms for the calibration curve and distribution of scores are well captured by simple GLM and beta models. Figures 3a,b show the resulting fits with parameters summarized in Appendix A. We observe significant skew in the score distribution which, as discussed in Section 1, poses a challenge to measuring calibration error with equal-width bins. We find that the dataset has more influence on the fits than the neural model, with ImageNet models the most similar to one another for a given data set, we consider ECE_{ebin} and ECE_{bias}. (Appendix B includes an analysis that varies the number of bins and finds that the optimal number of bins for bias minimization depends on the number of samples. This Appendix also includes calculations of variance across estimators, bin numbers, and sample sizes.)

Figure 4 plots the bias versus sample size for seven estimators, shown separately for each of three datasets. Because the curves for individual architectures look very similar to one another for a given data set, we have averaged over model architectures. The black dotted line indicates an unbiased estimator.

### 7 RESULTS

#### 7.1 Estimating bias on real models and data

Our bias-by-construction (BBC) framework uses the parametric fits to real models and datasets from Section 6 to estimate bias as follows. Each fit permits the analytical or numerical computation of TCE and can also be used in generative fashion to draw a synthetic set of examples. ECE can then be estimated from these samples, and the difference between the estimated ECE and TCE across many samples—1,000 in results to be presented—yields the bias (Equation 7).

We estimate bias for ECE_{ebin}, ECE_{bias}, and ECE_{sweep} using both equal-mass and equal-width binning, and also for the KDE estimator. Following Guo et al. (2017), we choose 15 bins for ECE_{ebin} and ECE_{bias}. (Appendix B includes an analysis that varies the number of bins and finds that the optimal number of bins for bias minimization depends on the number of samples. This Appendix also includes calculations of variance across estimators, bin numbers, and sample sizes.)

Figure 5 shows plots of the bias versus sample size for seven estimators, shown separately for each of three datasets. Because the curves for individual architectures look very similar to one another for a given data set, we have averaged over model architectures. The black dotted line indicates an unbiased estimator..

**Figure 3:** Maximum likelihood fits to empirical datasets illustrate large skew in their density distribution and calibration function. For each dataset, we fit (a) confidence distributions with Beta distribution and (b) calibration curves with generalized linear models across multiple model families, selecting the best model via the Akaike information criterion (details in Appendix A). We find the dataset source has a greater influence over the curves than the model architecture. (c) We plot the overall quality of the fits by computing the ECE_{ebin} averaged over 1000 simulated trials. Curves well-fit to the data lie close to the identity line.
我们增加的可能性是，我们生成了一个样本，该样本位于较低的bins中，由于其较低的样本密度，可能有一个较差的平均估计的TCE。在ImageNet中，由于置信度分布不太偏斜，EM和EW的差异仍然存在，但并不那么明显。

### 7.2 如何检测误校准？

从实用角度来看，从业者可能更关心的是能够回答一个直率性问题：该模型是否存在误校准？如果验证集提供了明确的证据存在误校准，那么必须采取进一步的步骤来纠正误校准。然而，由于ECE存在的偏差，仅仅观察到ECE > 0是不足以引起警惕的。

考虑的情况是一个模型的未知TCE，我们希望进行假设检验来确定我们是否可以拒绝零假设TCE=0。我们的能力来检测误校准取决于TCE、样本大小（n）和校准误差的估计方法。我们进行模拟，f(x) ∼ Beta(1, 1)和实际的校准曲线来自模型训练出CIFAR-10/100或ImageNet（见图3a或可靠性图和得分分布的Kängsepp, 2019）。

图5显示了类型II错误率为0.05（也被称为误报警率，或在完美校准的情况下未检测到误校准的失败率），对于常见的误差率（ECE_{EW BIN}）和在前一节中识别的最佳性能的误差率（ECE_{EM SWEEP}）。ECE_{EM SWEEP}得到显著更低的失败率，特别是对于小于10,000的样本。

### 图5

| TCE (%) | ECE_{EM SWEEP} | ECE_{EW BIN} |
|--------|----------------|--------------|
| 2%     | 0.25           | 0.30         |
| 4%     | 0.45           | 0.50         |
| 6%     | 0.65           | 0.70         |

我们注意到两个方法的局限性：要检测2%的误校准，需要超过10,000个样本；ECE_{EM SWEEP}特别对于10,000个样本，但ECE_{EM SWEEP}尤其对其10,000个样本，特别是对于10,000个样本。
and if one has under 500 samples, the miscalibration must be greater than 10% to be detected reliably.

### 7.3 Perfect calibration

In Section 7.1, we studied realistic scenarios of models whose outputs have the same statistics as common neural architectures on popular datasets. The BBC framework also allows us to simulate a continuum of models that differ systematically in TCE. For all metrics, bias increases as TCE decreases (details in Appendix C). This finding is not surprising because binned metrics always produce a nonnegative ECE estimate, and in the limit of a perfectly calibrated model, any deviation of the binning histogram from the diagonal will result in positive bias.

In this section, we compare the bias of estimators for the case of a perfectly calibrated model—the ultimate aim of designing methods that minimize miscalibration. To simulate perfect calibration, the calibration curve of the model is set to $E_Y[Y | f(X) = c] = c$, but we use the realistic confidence score distributions from the previous section.

Figure 6 illustrates the effect of sample size on bias for the seven different estimators under perfect calibration. Although the KDE estimator outperforms all others, it is not a viable candidate because it has a very high bias for realistic scenarios (Figure 1). Excluding KDE, ECE$^{EM}_{debiased}$ is the least biased metric, obtaining significantly lower bias than ECE$^{EM}_{SWEEP}$.

How do we reconcile these results with our previous finding (Figure 1) that ECE$^{EM}_{SWEEP}$ is preferred over ECE$^{EM}_{debiased}$? The present results assume a well calibrated model; the previous results are based on realistic scenarios. Whether one prefers ECE$^{EM}_{SWEEP}$ or ECE$^{EM}_{debiased}$ ultimately depends on a practitioner’s prior beliefs about a model’s degree of miscalibration. But to some degree we are splitting hairs: both ECE$^{EM}_{SWEEP}$ and ECE$^{EM}_{debiased}$ are consistently superior to common practice (ECE$^{EM}_{bin}$) and proposed improvements (e.g., ECE$^{EM}_{debiased}$) as recommended by Kumar et al. (2019).

### 8 DISCUSSION AND CONCLUSION

Calibration research typically focuses on recalibrating models, i.e., transforming $f(x)$ to $f'(x)$ (Platt et al. 1999; Zadrozny and Elkan 2001, 2002). We focus on estimating true calibration error, because without a good estimate, how is one to select and evaluate recalibration methods? The preferred recalibration method for a given model and data set is affected by bias: Table 1 shows that using ECE$^{EM}_{SWEEP}$ to select a recalibration method instead of ECE$^{EM}_{bin}$ leads to better choices and subsequently, better calibration on the test set. Indeed, bias may have impacted the conclusions of previous studies of calibration error, such as the well cited work of Guo et al. (2017). The choice of calibration error estimator can also impact the detection of miscalibration: Figure 5 indicates that ECE$^{EM}_{bin}$ is a more sensitive metric than ECE$^{EM}_{bin}$ for detecting if a model is miscalibrated.

Several authors attempt a different approach to recalibration: improving model calibration during training. For instance, Mukhoti et al. (2020) train a model with a batch size of 128 across multiple types of losses including maximum mean calibration error (Kumar et al. 2018) and Brier loss (Brier 1950) which explicitly minimizes calibration loss using 128 examples at a time. However, our results suggest that training a model with naive estimates of calibration error using a batch size $< O(1000)$ is a potentially flawed endeavor, particularly because the distribution of scores from the model changes throughout training, and any potential calibration measure may be more affected by the distribution of scores than the true calibration curve.

Our work can be extended in many directions which we did not have space here to consider, including: examining violations of our distributions assumptions and the setting where the confidence-score distributions are less skewed; studying the interplay between bias and vari-
We would like to thank Simon Kornblith, Jize Zhang, Tengyu Ma, and Ananya Kumar for helpful comments on this work.

References

Azulay, A. and Weiss, Y. (2018). Why do deep convolutional networks generalize so poorly to small image transformations? arXiv preprint arXiv:1805.12177.

Biggio, B. and Roli, F. (2018). Wild patterns: Ten years after the rise of adversarial machine learning. Pattern Recognition. https://arxiv.org/abs/1712.03141

Brier, G. W. (1950). Verification of forecasts expressed in terms of probability. Monthly weather review, 78(1):1–3.

Brocker, J. (2012). Estimating reliability and resolution of probability forecasts through decomposition of the empirical score. Climate Dynamics, 39:655–667.

Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., and Beijbom, O. (2020). nuScenes: A multimodal dataset for autonomous driving. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 11621–11631.

Chen, W., Sahiner, B., Samuelson, F., Pezeshk, A., and Petrick, N. (2018). Calibration of medical diagnostic classifier scores to the probability of disease. Statistical methods in medical research, 27(5):1394–1409.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). ImageNet: A Large-Scale Hierarchical Image Database. In CVPR09.

Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., and Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. nature, 542(7639):115–118.

Esteva, A., Robicquet, A., Ramsundar, B., Kuleshov, V., DePristo, M., Chou, K., Cui, C., Corrado, G., Thrun, S., and Dean, J. (2019). A guide to deep learning in healthcare. Nature medicine, 25(1):24–29.

Ferro, C. A. T. and Fricker, T. E. (2012). A bias-corrected decomposition of the brier score. Quarterly Journal of the Royal Meteorological Society, 138:1954–1960.

Geiger, A., Lenz, P., Stiller, C., and Urtasun, R. (2013). Vision meets robotics: The kitti dataset. The International Journal of Robotics Research, 32(11):1231–1237.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (2004). Bayesian Data Analysis. Chapman and Hall/CRC, 2nd ed. edition.

Gneiting, T. and Vogel, P. (2018). Receiver operating characteristic (roc) curves. arXiv preprint arXiv:1809.04808.

Green, D. M., Swets, J. A., et al. (1966). Signal detection theory and psychophysics, volume 1. Wiley New York.

Gulshan, V., Peng, L., Coram, M., Stumpe, M. C., Wu, D., Narayanaswamy, A., Venugopalan, S., Widner, K., Madams, T., Cuadros, J., et al. (2016). Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. Jama, 316(22):2402–2410.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. (2017). On calibration of modern neural networks. International Conference on Machine Learning (ICML).

Gupta, K., Rahimi, A., Ajanthan, T., Mensink, T., Sminchisescu, C., and Hartley, R. (2020). Calibration of neural networks using splines. arXiv preprint arXiv:2006.12800.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In Computer Vision and Pattern Recognition (CVPR).

Hendrycks, D. and Dietterich, T. (2019). Benchmarking neural network robustness to common corruptions and perturbations. In International Conference on Learning Representations (ICLR). https://arxiv.org/abs/1807.01697

Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. (2017). Densely connected convolu-
Mitigating Bias in Calibration Error Estimation

Huang, G., Sun, Y., Liu, Z., Sedra, D., and Weinberger, K. Q. (2016). Deep networks with stochastic depth. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708. Springer.

Karanikolakis, A., Cain, N., Tran, D., Lakshminarayanan, B., Shlens, J., Mozer, M. C., and Roelofs, B. (2021). Soft calibration objectives for neural networks. In *Advances in Neural Information Processing Systems*.

Krishnan, R. and Tickoo, O. (2020). Improving model calibration with accuracy versus uncertainty optimization. *ArXiv*, abs/2012.07923.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2018). Focal loss for dense object detection.

Mukhoti, J., Kulharia, V., Sanyal, A., Golodetz, S., Torr, P. H., and Dokania, P. K. (2020). Calibrating deep neural networks using focal loss. *arXiv preprint arXiv:2002.09437*.

Naeini, M. P., Cooper, G. F., and Hauskrecht, M. (2015). Obtaining well calibrated probabilities using bayesian binning. In *AAAI Conference on Artificial Intelligence*. NIH Public Access.

Nixon, J., Dusenberry, M. W., Zhang, L., Jerfel, G., and Tran, D. (2019). Measuring calibration in deep learning. In *CVPR Workshops*, pages 38–41.

Pesce, L. L., Metz, C. E., and Berbaum, K. S. (2010). On the convexity of roc curves estimated from radiological test results. *Academic radiology*, 17(8):960–968.

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.

Recht, B., Roelofs, R., Schmidt, L., and Shankar, V. (2019). Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning*, pages 5389–5400.

Sun, P., Kretzschmar, H., Doshi-Velez, F., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., et al. (2020). Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2446–2454.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I. J., and Fergus, R. (2013). Intriguing properties of neural networks. In *International Conference on Learning Representations (ICLR)*. *http://arxiv.org/abs/1312.6199*.

Widmann, D., Lindsten, F., and Zachariah, D. (2019). Calibration tests in multi-class classification: A unifying framework. *NeurIPS*.

Wu, Y., Jiang, X., Kim, J., and Ohno-Machado, L. (2012). I-spline smoothing for calibrating predictive models. *AMIA Summits on Translational Science Proceedings*, 2012:39.

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

Zadrozny, B. and Elkan, C. (2002). Transforming classifier scores into accurate multiclass probability estimates. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Zagoruyko, S. and Komodakis, N. (2016). Wide residual networks. *arXiv preprint arXiv:1605.07146*.

Zhang, J., Kaiikura, B., and Han, T. (2020). Mix-n-match: Ensemble and compositional methods for uncertainty calibration in deep learning. *arXiv preprint arXiv:2003.07329*.
A Maximum-likelihood fits

A.1 Confidence score distribution fits

Table 2 provides parameters of best fit for the Beta distribution for each of 10 empirical datasets, obtained by fitting the top-label confidence score via maximum likelihood estimation.

Table 2: Parameters of best fit for Beta distribution investigated in Section 6.

| Dataset             | $\hat{\alpha}$ | $\hat{\beta}$ |
|---------------------|-----------------|----------------|
| resnet110_c10       | 2.7752          | 0.0478         |
| resnet110_SD_c10    | 2.1714          | 0.0394         |
| resnet_wide32_c10   | 2.3806          | 0.0379         |
| densenet40_c10      | 1.9824          | 0.0397         |
| resnet110_c100      | 1.1823          | 0.1081         |
| resnet110_SD_c100   | 1.1233          | 0.1147         |
| resnet_wide32_c100  | 1.0611          | 0.0650         |
| densenet40_c100     | 1.0805          | 0.0808         |
| resnet152_imgnet    | 1.1359          | 0.2069         |
| densenet161_imgnet  | 1.1928          | 0.2206         |

Global optimia $\hat{\alpha} \in [0, 200]$, $\hat{\beta} \in [0, 50]$ are approximately computed using a recursively-refining brute-force search until both parameters are established to within an absolute tolerance of $1 \times 10^{-5}$. Each step in the recursion contracts a linear sampling grid ($N = 11$) by a factor of $\gamma = .5$ centered on the previously established optimal parameter, subject to the constraints $\alpha, \beta > 0$. Experiments confirmed that the computed optima were robust to the hyperparameters $N, \gamma$.

$$\arg\min_{\alpha, \beta} \sum_i -\ln \frac{x_i^{\alpha-1}(1-x_i)^{\beta-1}}{\frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}}$$  \hspace{1cm} (9)

A.2 Calibration curve fits

Table 2 provides parameters fit to calibration functions. For each sample image $x_i$ in the image dataset, define $s_i = f(x_i)$ to be the score (the output of the top-scoring logit after softmax) and $y_i \in \{0, 1\}$ to be the classification ($y_i = 1$ when the top-scoring logit correctly classified image $x_i$) for the sample image. The loss for the binary generalized linear model (GLM) across different combinations of link functions $g(y)$ and transform functions $t(s)$ was optimized via the standard loss (Gelman et al., 2004):

$$\arg\min_{b_0, b_1} \sum_i -\ln p_i^{y_i}(1-p_i)^{1-y_i}, \quad p_i = g^{-1}(b_0 + b_1 t(s_i))$$  \hspace{1cm} (10)

For each dataset, the GLM of best fit was selected via the Akaike Information Criteria using the likelihood at the optimized parameter values.

A.3 Comparing ECE$_{\text{bin}}$ computed on simulated data versus real data

In Figure 3c, we compare the ECE$_{\text{bin}}$ computed on the original logit output of each model to the average ECE$_{\text{bin}}$ we obtain after sampling 1,000 simulated datasets from our parametric fits. Table 3 reports the ECE$_{\text{bin}}$ measurements that we plot in Figure 3. We observe that the two measurements of ECE$_{\text{bin}}$ are relatively close, indicating that our parametric models are well-fit to the original data.

In Figure 3c, we compare the ECE$_{\text{bin}}$ computed on the original logit output of each model to the average ECE$_{\text{bin}}$ we obtain after sampling 1,000 simulated datasets from our parametric fits. Table 3 reports the ECE$_{\text{bin}}$ measurements that we plot in Figure 3. We observe that the two measurements of ECE$_{\text{bin}}$ are relatively close, indicating that our parametric models are well-fit to the original data.
Table 3: Parameters of best fit for calibrations functions investigated in Section 6 (table continues on next page).

| dataset_name    | glm_name               | AIC   | $b_0$ | $b_1$    |
|-----------------|------------------------|-------|-------|----------|
| resnet110_c10   | logflip_logflip_b0_b1 | 2779.22 | -0.24 | 0.30     |
|                 | logit_logflip_b0_b1    | 2790.40 | -0.55 | -0.38    |
|                 | logit_logflip_b1       | 2827.51 | -0.31 |          |
|                 | logit_logit_b0_b1      | 2840.70 | -0.38 | 0.36     |
|                 | logit_logit_b1         | 2900.02 | 0.30  |          |
|                 | logflip_logflip_b1     | 2932.09 | 0.34  |          |
|                 | log_log_b0_b1          | 3221.72 | -0.06 | 2.53     |
|                 | logit_logit_b0         | 3799.63 | 1.99  |          |
|                 | logflip_logflip_b0     | 3811.98 | -2.13 |          |
|                 | log_log_b0             | 3829.05 | -0.13 |          |
|                 | logit_logflip_b0       | 3868.40 | 1.95  |          |
|                 | log_log_b1             | 4281.78 | 4.75  |          |
| resnet110_SD_c10| logit_logflip_b0_b1    | 2498.98 | -0.27 | -0.35    |
|                 | logit_logit_b1         | 2502.52 | 0.30  |          |
|                 | logit_logflip_b1       | 2508.70 | 0.30  |          |
|                 | logit_logit_b0_b1      | 2538.41 | -0.26 | 0.33     |
|                 | logflip_logflip_b0_b1  | 2550.29 | -0.36 | 0.27     |
|                 | logflip_logflip_b1     | 2572.85 | 0.35  |          |
|                 | log_log_b0_b1          | 2594.91 | -0.08 | 1.98     |
|                 | log_log_b0             | 3137.19 | -0.19 |          |
|                 | logflip_logflip_b0     | 3150.42 | -1.80 |          |
|                 | logit_logit_b0         | 3175.58 | 1.58  |          |
|                 | logit_logflip_b0       | 3179.67 | 1.56  |          |
|                 | log_log_b1             | 3697.37 | 3.77  |          |
| resnet_wide32_c10| logit_logit_b1        | 2483.34 | 0.26  |          |
|                 | logflip_logflip_b0_b1  | 2487.69 | -0.47 | 0.22     |
|                 | logit_logit_b0_b1      | 2511.39 | -0.13 | 0.28     |
|                 | logit_logflip_b0_b1    | 2558.45 | -0.26 | -0.28    |
|                 | logit_logflip_b1       | 2586.47 | -0.25 |          |
|                 | log_log_b0_b1          | 2647.03 | -0.12 | 1.87     |
|                 | logflip_logflip_b1     | 2713.17 | 0.30  |          |
|                 | log_log_b0             | 2981.24 | -0.21 |          |
|                 | logflip_logflip_b0     | 2983.05 | -1.70 |          |
|                 | logit_logit_b0         | 2989.90 | 1.49  |          |
|                 | logit_logflip_b0       | 3055.55 | 1.45  |          |
|                 | log_log_b1             | 4582.09 | 4.61  |          |
| densenet40_c10  | logit_logflip_b1       | 2910.62 | -0.26 |          |
|                 | logit_logit_b0_b1      | 2961.31 | -0.40 | 0.31     |
|                 | logit_logflip_b0_b1    | 3000.23 | -0.38 | -0.31    |
|                 | logit_logit_b0         | 3001.78 | -0.31 | 0.24     |
|                 | logit_logit_b1         | 3021.34 | 0.25  |          |
|                 | logflip_logflip_b1     | 3027.78 | 0.31  |          |
|                 | log_log_b0_b1          | 3153.38 | -0.12 | 2.04     |
|                 | log_log_b0             | 3531.22 | -0.22 |          |
|                 | logflip_logflip_b0     | 3589.11 | -1.60 |          |
|                 | logit_logit_b0         | 3601.85 | 1.37  |          |
|                 | logit_logflip_b0       | 3679.95 | 1.30  |          |
|                 | log_log_b1             | 4735.18 | 4.27  |          |
| resnet110_c100  | logflip_logflip_b0_b1  | 8181.97 | -0.11 | 0.28     |
|                 | logit_logit_b0_b1      | 8206.19 | -0.88 | 0.39     |
|                 | logflip_logflip_b1     | 8301.28 | 0.31  |          |
|                 | logit_logit_b0_b1      | 8371.33 | -1.01 | -0.40    |
|                 | logit_logit_b1         | 8732.11 | 0.25  |          |
|                 | log_log_b0_b1          | 8918.21 | -0.16 | 2.35     |
|                 | logit_logflip_b1       | 8926.99 | -0.23 |          |
|                 | logit_logflip_b0       | 10903.83 | 0.74 |          |
|                 | logit_logit_b0         | 10943.95 | 0.72 |          |
|                 | logit_logflip_b0       | 10964.91 | -1.12 |          |
|                 | log_log_b0             | 11002.20 | -0.40 |          |
|                 | log_log_b1             | 11850.89 | 4.26 |          |
| dataset_name | glm_name | AIC    | $b_0$ | $b_1$ |
|-------------|----------|--------|-------|-------|
| resnet110_SD_c100 | logit_logit_b0_b1 | **7873.61** | -0.88 | 0.49 |
|             | logflip_logflip_b0_b1 | 7878.19 | -0.09 | 0.35 |
|             | logflip_logflip_b1    | 7932.28 | 0.38  |       |
|             | logit_logflip_b0_b1   | 7944.61 | -1.04 | -0.52 |
|             | logit_logit_b1        | 8315.51 | 0.32  |       |
|             | log_log_b0_b1         | 8437.82 | -0.11 | 2.18  |
|             | logit_logflip_b1      | 8510.36 | -0.30 |       |
|             | log_log_b1            | 9988.07 | 3.30  |       |
|             | logit_logit_b0        | 10034.03 | 0.80  |       |
|             | log_log_b0            | 10810.90 | -0.37 |       |
|             | logflip_logflip_b1    | 10823.15 | -1.16 |       |
|             | logit_logflip_b0      | 10834.48 | 0.78  |       |
| resnet_wide32_c100 | logflip_logflip_b0_b1 | **7183.93** | -0.13 | 0.21 |
|             | logit_logit_b0_b1     | 7219.14 | -0.98 | 0.33  |
|             | logflip_logflip_b1    | 7233.51 | 0.25  |       |
|             | logit_logflip_b0_b1   | 7297.00 | -1.06 | -0.34 |
|             | logit_logit_b1        | 7626.21 | 0.19  |       |
|             | log_log_b0_b1         | 7650.97 | -0.24 | 2.51  |
|             | logit_logflip_b1      | 7795.28 | -0.17 |       |
|             | logflip_logflip_b0    | 8077.39 | 0.40  |       |
|             | logit_logflip_b0      | 8097.38 | 0.49  |       |
|             | log_log_b0            | 9000.24 | -0.49 |       |
|             | logit_logit_b0        | 9009.51 | 0.49  |       |
|             | log_log_b1            | 11911.51 | 5.48  |       |
| densenet40_c100  | logflip_logflip_b0_b1 | **8158.28** | -0.97 | 0.34 |
|             | logit_logit_b0_b1     | 8187.43 | -0.12 | 0.22  |
|             | logflip_logflip_b0_b1 | 8267.77 | -1.08 | -0.35 |
|             | logit_logflip_b1      | 8368.86 | 0.25  |       |
|             | logit_logit_b1        | 8783.50 | 0.19  |       |
|             | log_log_b0_b1         | 8832.20 | -0.25 | 2.26  |
|             | logit_logflip_b1      | 8918.57 | 0.18  |       |
|             | logit_logit_b0        | 10134.24 | 0.47  |       |
|             | logit_logflip_b0      | 10182.61 | 0.45  |       |
|             | log_log_b0            | 10242.15 | -0.94 |       |
|             | log_log_b0            | 10261.01 | -0.50 |       |
|             | log_log_b1            | 11191.51 | 5.48  |       |
|             | logit_logit_b0_b1     | **8158.28** | -0.97 | 0.34 |
|             | logflip_logflip_b0_b1 | 8229.43 | -0.12 | 0.22  |
|             | logit_logflip_b0_b1   | 8267.77 | -1.08 | -0.35 |
|             | logit_logflip_b1      | 8368.86 | 0.25  |       |
|             | logit_logit_b1        | 8783.50 | 0.19  |       |
|             | log_log_b0_b1         | 8832.20 | -0.25 | 2.26  |
|             | logit_logflip_b1      | 8918.57 | 0.18  |       |
|             | logit_logit_b0        | 10134.24 | 0.47  |       |
|             | logit_logflip_b0      | 10182.61 | 0.45  |       |
|             | log_log_b0            | 10242.15 | -0.94 |       |
|             | log_log_b0            | 10261.01 | -0.50 |       |
|             | log_log_b1            | 11191.51 | 5.48  |       |
| resnet152_imgnet | logflip_logflip_b0_b1 | **18729.85** | -0.12 | 0.58 |
|             | logit_logit_b0_b1     | 18783.22 | -0.29 | 0.65  |
|             | log_log_b0_b1         | 18785.44 | -0.03 | 1.32  |
|             | logflip_logflip_b1    | 18872.14 | 0.65  |       |
|             | logit_logit_b1        | 19074.37 | 0.57  |       |
|             | logit_logflip_b0_b1   | 19095.40 | -0.82 | -0.79 |
|             | log_log_b1            | 19840.25 | 1.53  |       |
|             | logit_logflip_b1      | 20062.10 | -0.50 |       |
|             | log_log_b0            | 20935.09 | -1.41 |       |
|             | logit_logit_b0        | 20968.50 | -0.28 |       |
|             | log_log_b0            | 27012.77 | 1.12  |       |
|             | logit_logit_b0        | 27094.11 | 1.11  |       |
| densenet161_imgnet | log_log_b0_b1 | **18202.41** | -0.03 | 1.27 |
|             | logit_logit_b0_b1     | 18460.70 | -0.25 | 0.68  |
|             | logflip_logflip_b1    | 18521.48 | 0.67  |       |
|             | logit_logit_b0_b1     | 18534.07 | -0.10 | 0.61  |
|             | logit_logit_b1        | 18822.25 | 0.60  |       |
|             | logit_logflip_b0_b1   | 18913.25 | -0.77 | -0.80 |
|             | log_log_b1            | 19403.85 | 1.44  |       |
|             | logit_logflip_b1      | 19562.58 | -0.54 |       |
|             | logit_logit_b0        | 26426.38 | 1.19  |       |
|             | logflip_logflip_b0    | 26445.91 | -1.46 |       |
|             | logit_logit_b0        | 26519.76 | 1.18  |       |
|             | log_log_b0            | 26662.65 | -0.27 |       |
Table 4: \( \text{ECE}_{\text{bin}} \) reported in Figure 3c.

| Model               | \( \text{ECE}_{\text{bin}} \) (%) | \( \langle \text{ECE}_{\text{bin}} \rangle \) (%) simulated |
|---------------------|----------------------------------|--------------------------------------------------------|
| resnet110_c10       | 6.67                             | 8.42                                                   |
| resnet110_SD_c10    | 6.54                             | 8.79                                                   |
| resnet_wide32_c10   | 6.09                             | 8.44                                                   |
| densenet40_c10      | 6.70                             | 8.09                                                   |
| resnet110_c100      | 20.26                            | 18.87                                                  |
| resnet110_SD_c100   | 17.44                            | 15.78                                                  |
| resnet_wide32_c100  | 20.40                            | 17.53                                                  |
| densenet40_c100     | 23.12                            | 19.69                                                  |
| resnet152_imgnet    | 6.85                             | 9.26                                                   |
| densenet161_imgnet  | 6.15                             | 6.87                                                   |
B Bias and variance in calibration metrics

B.1 Bias

We evaluate bias for various calibration metrics using both equal-width and equal-mass binning as we vary both the sample size \( n \) and the number of bins \( b \). These plots should be seen as an alternative visualization to \( 3 \) where we additionally compare to different choices for the fixed number of bins \( b \). Since the ECE_{SWEEP} metrics adaptively choose a different number of bins for each sample size, we display the bin number for this metric as \(-1\).

We find that ECE_{BIN} can overestimate the true calibration error and there exists an optimal number of bins that produces the least biased estimator that changes with the number of samples \( n \). Additionally, equal mass binning generally results in a less biased metric than equal width binning.

CIFAR-10 ResNet-110. Figure 7 assume parametric curves for \( p(f(x)) \) and \( \mathbb{E}_Y[Y \mid f(X) = c] \) that we obtain from maximum-likelihood fits to CIFAR-10 ResNet-110 model output.

CIFAR-100 Wide ResNet-32. Figure 8 assume parametric curves for \( p(f(x)) \) and \( \mathbb{E}_Y[Y \mid f(X) = c] \) that we obtain from maximum-likelihood fits to CIFAR-100 Wide ResNet-32 model output.

ImageNet ResNet-152. Figure 9 assume parametric curves for \( p(f(x)) \) and \( \mathbb{E}_Y[Y \mid f(X) = c] \) that we obtain from maximum-likelihood fits to ImageNet ResNet-152 model output.

---

Figure 7: Bias for various calibration metrics assuming curves fit to CIFAR-10 ResNet-110 output. We plot bias for various calibration metrics using both equal-width binning (left column) and equal-mass binning (right column) as we vary both the sample size \( n \) and the number of bins \( b \).
Figure 8: Bias for various calibration metrics assuming curves fit to CIFAR-100 Wide ResNet-32 output. We plot bias for various calibration metrics using both equal-width binning (left column) and equal-mass binning (right column) as we vary both the sample size $n$ and the number of bins $b$.

Figure 9: Bias for various calibration metrics assuming curves fit to ImageNet ResNet-152 output. We plot bias for various calibration metrics using both equal-width binning (left column) and equal-mass binning (right column) as we vary both the sample size $n$ and the number of bins $b$. 
B.2 Variance

We also compute the variance for various calibration metrics using both equal-width and equal-mass binning as we vary both the sample size $n$ and the number of bins $b$. As expected, the variance decreases with number of samples, but, unlike the bias, there is no clear dependence on the number of bins.

CIFAR-10 ResNet-110. Figure 10 assume parametric curves for $p(f(x))$ and $\mathbb{E}[Y \mid f(X) = c]$ that we obtain from maximum-likelihood fits to CIFAR-10 ResNet-110 model output.

CIFAR-100 Wide ResNet-32. Figure 11 assume parametric curves for $p(f(x))$ and $\mathbb{E}[Y \mid f(X) = c]$ that we obtain from maximum-likelihood fits to CIFAR-100 Wide ResNet-32 model output.

ImageNet ResNet-152. Figure 12 assume parametric curves for $p(f(x))$ and $\mathbb{E}[Y \mid f(X) = c]$ that we obtain from maximum-likelihood fits to ImageNet ResNet-152 model output.

![Figure 10: $\sqrt{\text{Variance}}$ for various calibration metrics assuming curves fit to CIFAR-10 ResNet-110 output. We plot $\sqrt{\text{Variance}}$ for various calibration metrics using both equal-width binning (left column) and equal-mass binning (right column) as we vary both the sample size $n$ and the number of bins $b$.](image-url)
Table 1: Variance for various calibration metrics assuming curves fit to CIFAR-100 Wide ResNet-32 output. We plot √Var for various calibration metrics using both equal-width binning (left column) and equal-mass binning (right column) as we vary both the sample size $n$ and the number of bins $b$.  

Table 2: Variance for various calibration metrics assuming curves fit to ImageNet ResNet-152 output. We plot √Var for various calibration metrics using both equal-width binning (left column) and equal-mass binning (right column) as we vary both the sample size $n$ and the number of bins $b$. 

![Table 1](image1.png)  

![Table 2](image2.png)
C Controlling true calibration error using BBC

We evaluate the estimation bias of calibration estimators as we systematically vary the TCE. Figure 13 shows the average estimated calibration error for $\text{ECE}_{\text{bin}}^{\text{EW}}$ and $\text{ECE}_{\text{sweep}}^{\text{EM}}$ versus the TCE. The average calibration error is computed across $m = 1,000$ simulated datasets, and we include results for two sample sizes, $n = 200$ and $n = 5,000$, and two score distributions, $f(x) \sim \text{Uniform}(0, 1)$ and $f(x) \sim \text{Beta}(1.1, 0.1)$, the beta distribution fit to the CIFAR-100 Wide ResNet_32. To control the TCE, we assume $\mathbb{E}_Y[Y \mid f(X) = c] = c^d$ and vary $d \in [1, 10]$. When $d = 1$ the true calibration curve is $\mathbb{E}_Y[Y \mid f(X) = c] = c$, which means the model’s predicted confidence score is exactly equal to its empirical accuracy and thus the TCE is 0%. As we increase $d$, we move the true calibration curve farther away from the perfect calibration curve, which increases the TCE of the model.

The estimation bias can be seen visually as the difference between the ECE and the $y = x$ line. Perfect estimation (0 bias) corresponds to the $y = x$ line. Bias is highest when the model is perfectly calibrated (TCE is 0%) and generally decreases as TCE increases. A larger sample size of $n = 5,000$ reduces the bias, but with perfectly calibration ECE$_{\text{bin}}$ can still be off by 2%. The ECE$_{\text{sweep}}^{\text{EM}}$ metric significantly reduces this bias.

D What number of bins does $\text{ECE}_{\text{sweep}}^{\text{EM}}$ choose?

For Figure 14, the uncalibrated plot assumes $\mathbb{E}_Y[Y \mid f(X) = c] = \text{logistic}(10 * c - 5)$ while the calibrated plot assumes $\mathbb{E}_Y[Y \mid f(X) = c] = c$. Both experiments assume $f(x) \sim \text{Uniform}(0, 1)$.

Figure 13: Bias in calibration estimation increases as TCE decreases. Average ECE (%) for $\text{ECE}_{\text{bin}}^{\text{EW}}$ (left) and $\text{ECE}_{\text{sweep}}^{\text{EM}}$ (right) versus the TCE (%), with varying sample size and score distributions. The estimator bias is systematically worse for better calibrated models, and the effect is more egregious with fewer samples. At $n = 200$ samples, depending on the score distribution, an ECE$_{\text{bin}}^{\text{EW}}$ estimate of 12% could either correspond to 5% or 8% TCE. ECE$_{\text{sweep}}^{\text{EM}}$ somewhat mitigates the bias and ambiguity in calibration error estimation.

Figure 14: Bins chosen by equal mass ECE$_{\text{sweep}}^{\text{EM}}$ method. We plot equal mass ECE$_{\text{bin}}^{\text{EM}}$ % versus number of bins for various sample sizes $n$. We highlight the TCE with a horizontal dashed line and show the average number of bins chosen by the ECE$_{\text{sweep}}^{\text{EM}}$ method for different sample sizes with vertical dashed lines. When the model is uncalibrated (left) ECE$_{\text{sweep}}^{\text{EM}}$ chooses a bin number that is close to optimal. However, for perfectly calibrated models (right), the optimal number of bins is small ($<4$), and ECE$_{\text{sweep}}^{\text{EM}}$ does not do a good job of selecting a good bin number. The incorrect bin selection may partially explain why ECE$_{\text{sweep}}^{\text{EM}}$ still has some bias for perfectly calibrated models. However, we note that any binning-based technique that always outputs a positive number will never be completely unbiased for perfectly calibrated models.